Ocean Engineering 93 (2015) 25–44

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng

Visual sensing for autonomous underwater exploration and intervention tasks ☆

Francisco Bonin-Font [a,*], Gabriel Oliver [a], Stephan Wirth [a], Miquel Massot [a], Pep Lluis Negre [a], Joan-Pau Beltran [b]

[a] Department of Mathematics and Computer Science; Systems, Robotics and Vision Group, University of the Balearic Islands, Carretera de Valldemossa km 7.5, Anselm Turmeda building, Palma de Mallorca, Spain
[b] SOCIB, Balearic Islands Coastal Observing and Forecasting System; Parc Bit, Naorte, Bloc A 2p. pta. 3. Palma de Mallorca Spain

ARTICLE INFO

Article history:
Received 8 November 2013
Accepted 5 November 2014
Available online 25 November 2014

Keywords:
Underwater intervention and exploration
Vision-based navigation
Visual object detection and tracking

ABSTRACT

Underwater activities, such as surveying or interventions, carried out by autonomous robots, can benefit greatly from using a vision system. Optics based systems provide information at a spatial and temporal resolution higher than their acoustic counterparts. At present, they are the best option when high precision maneuvering and manipulation is needed, if there is good visibility. This paper presents a new system designed to provide visual information in submarine tasks such as navigation, surveying, mapping and intervention. The main advantages of our system, called Fugu-f (Fugu flexible), are its robustness in both the mechanical structure and the software components, its flexibility, since it is installed as an external module and is adaptable to different vehicles and missions, and its capacity to operate in real-time. Experiments of surveying and object manipulation carried out in real conditions in the context of the TRIDENT project show the suitability of the system and its scientific and industrial potential applications.

1. Introduction and related work

Oceans are an attractive environment for humans since they offer an enormous quantity of food, minerals and energy resources. Underwater research and development has evolved rapidly during the last decades, but oceans are still a challenging media due to the extreme conditions at which humans and machines have to operate.

Exploration and intervention are becoming two fundamental tasks for many underwater industrial and scientific applications: construction or maintenance of underwater infrastructures, object search and identification, wreck retrieval, mapping of underwater regions or rescue missions. All these applications often entail manipulation activities such as opening and closing valves, pushing buttons, object recovery, hooking, cutting, drilling, pulling or pushing material. Currently, since operating at great depths prevents us from using human divers, companies and research institutions usually use ROVs (Remotely Operated Vehicle) or crewed vehicles to perform all the aforementioned activities, increasing operational costs and human security protocols (Bagley et al., 2007).

AUVs (Autonomous Underwater Vehicles) are a promising tool for underwater missions, especially if they can be equipped with manipulators (so-called I-AUVs or Intervention AUVs). In this way, mission costs can be reduced and human intervention in critical situations (deep ocean, under ice, mine retrieval or operating in classified areas) can be minimized, automatizing as many procedures as possible.

The fact that these tasks need to be performed as efficiently and accurately as possible has induced many researchers to continuously improve manipulators, mechatronics and controllers (Wilson et al., 2011; Ishitsuka and Ishii, 2005). However, accuracy in manipulation depends not only on the manipulator itself and its control, but also on the performance of the robot sensorial equipment and the algorithms used to process their readings.

Traditionally, autonomous underwater vehicles base their navigability and effectiveness mostly on acoustic sensors such as DVLs or sonars. Besides, the low bandwidth and the significant time delay inherent to acoustic subsea communications represent a considerable pitfall to remotely operate a manipulator and make

☆This work is partially supported by the Spanish Ministry of Economy and Competitiveness under Contracts PTA2011-05077 and DPI2011-27977-C03-02, FEDER Fundings, Govern Balear (Ref 71/2011) and by the European Commissions FP7 under Grant agreement 248497 (TRIDENT Project)

* Corresponding author. Tel.: +34 971171391.
E-mail addresses: francisco.bonin@uib.es (F. Bonin-Font), goliver@uib.es (G. Oliver), stephan.wirth@gmail.es (S. Wirth), miquel.massot@uib.es (M. Massot), pl.negre@uib.es (P. Lluis Negre), joanpau.beltran@uib.es (J.-P. Beltran).

http://dx.doi.org/10.1016/j.oceaneng.2014.11.005
0029-8018/

it more complicated for controllers to react on time when difficulties arise (Marani et al., 2009).

Vision is widely accepted as one of the most powerful and useful sensors in robotics, outperforming sonar, or laser range finders, in terms of temporal or spatial resolution, specially in water. Subsea environments have particular characteristics and present specific problems (light attenuation and scattering, non-uniform lighting and shadows, suspended particles or abundant marine life) that have to be taken into account during image acquisition for efficient, reliable, and robust underwater image processing (Bonin-Font et al., 2011).

Nowadays, few AUVs are equipped with manipulators and cameras, and only a couple of them have shown field capabilities applicable to offshore industry or for rescue and recovery missions.

One of the pioneers in integrating a manipulator in an I-AUV was the VORTEX/PA10 (Rigaud et al., 1998), a robot with 5 DOF (Degrees of Freedom) operated as an AUV carrying a 7 DOF PA10 arm. Later, Evans et al. (2003) projected ALIVE, an AUV with 4 DOF with an attached manipulator of 7 DOF and equipped with a camera used for approaching and docking to a fixed panel where the manipulation had to be carried out. SAUVIM (Kim and Yuh, 2004) is an AUV carrying a 7 DOF electrical driven arm (ANSALDO), initially designed to recover test missiles from the seabed in Hawaii. SAUVIM is also equipped with a camera to help in the manipulation process. Several successful tests in underwater scenarios were carried out in the SAUVIM project, increasing the expectations in these kinds of systems.

In this context, the TRIDENT project (Sanz et al., 2010) developed under the FP7 European framework proposes a new methodology for multipurpose underwater intervention, applicable in multiple scopes and that goes beyond the methods existing nowadays. This project provides a new and successfully proven method to identify and recover objects of interest in the sea bottom using an AUV with manipulation capabilities and guided by visual sensors. The originality of this project is in the whole engineered process to define, identify and manipulate different targets located in diverse underwater environments.

In TRIDENT, missions are run in 2 stages, a first one, where the robot aided by an ASC (Autonomous Surface Craft) explores and maps an unknown environment and provides video sequences to a human operator who views them and identifies/marks where is the object to be manipulated/recovered. In a second stage, the AUV navigates autonomously the previously generated map to localize the scene in which the object selected by the operator is laying. Once the object has been recognized, the AUV stays on it to perform the manipulation task. Although the strongest novelty lies in the whole intervention process, this paper is focused on its particular visual system that makes it possible.

The sensorial capabilities of the AUV used in the project were enriched with a new flexible and configurable external stereo vision module called Fugu-f, developed by the Systems, Robotics and Vision Group of the University of the Balearic Islands, and used for many of the tasks included in the AUV missions: surveying, registering, mapping and reconstructing the environment, navigating, identifying the target and assisting the manipulation tasks (target tracking and station keeping). Fugu-f is prepared to work up to 250 m depth and provides multiple visual facilities such as image/video grabbing, visual odometry, 3D reconstruction, object detection/tracking and assistance to manipulation tasks.

Fugu-f applied to an I-AUV has several advantages with respect to the state of the art visual solutions for underwater surveying and manipulation:

1. Traditionally, cameras and their electronics supports for observation underwater were mounted in rigid structures (Shortis et al., 2010) or usually integrated in underwater vehicles from early on in the design, having specific and reserved room in the hull and the electronics compartment (Dunbabin et al., 2005; Hildebrandt et al., 2012; Williams et al., 2008). Changes in the vehicle design can lead to changes in the visual system design or location, and vice versa. However, similar to Hogue et al. (2006), Fugu-f is an autonomous external module with high adaptability and flexibility. It can be installed in any underwater vehicle with an external power supply and a reduced payload available. The module incorporates two stereo rigs which can be placed in multiple configurations, depending on the mission to be carried out. Changes in the vehicle would not affect the visual system and changes in the Fugu-f do not affect the vehicle.
2. All the system components are external, guaranteeing their watertightness to 250 m depth.
3. The software installed in the system can also be adapted to the needs of the host vehicle. The system has wired ethernet communication capabilities, which permit its link with any other computer system installed in the host vehicle.

To the best of our knowledge, there are no other external visual systems in the literature with the same characteristics, and specially addressed to underwater vehicles, that have been successfully tested in autonomous underwater manipulation tasks.

A comparative of the different systems cited above is shown is Table 1.

From the point of view of the particular application to identify autonomously a sunken object of interest, the main contribution of this work lies in the process of defining the object model and the algorithm for its recognition. The object model cannot be defined

Table 1
Comparative of different underwater visual systems.

System	Adaptability	Number of cameras	Vision-based Applications	Depth Range
VORTEX/PA10 (Rigaud et al., 1998)	Internal, non-adaptable	Monocular	Scene analysis, pipe racking	Shallow waters
ALIVE (Evans et al., 2003)	Internal, non-adaptable	Monocular	Localization, docking, aiding to the manipulation task	<300 m
SAUVIM (Kim and Yuh, 2004)	Internal, non-adaptable	6 Monocular cameras	Observation and target localization	<6000
(Shortis et al., 2010)	Rigid, towed	1 Stereo rig + 2 still cameras	Habitat mapping and biodiversity surveying	<1500 m
(Dunbabin et al., 2005)	Internal, non-adaptable	2 Stereo rigs	Image grabbing, observation and localization	<100 m
(Hildebrandt et al., 2012)	Internal, non-adaptable	1 Stereo rigs	Localization and mapping	<150 m
AQUASENSOR (Hogue et al., 2006)	External, non-adaptable, single configuration	1 Stereo rigs	Image grabbing and offline 3D reconstruction	Shallow waters
Fugu-f	External, adaptable to multiple vehicles and configurations	2 Stereo rigs	Image grabbing, localization, online 3D reconstruction, scene/object recognition, docking, aiding to the manipulation task	<250 m

previously because the target appearance in the scene is unknown a priori, thus the system does not depend on any external model database. Instead of that, the model is generated from few selected images recorded during the exploration stage. As a consequence, the model can include not only the object but also the part of its surroundings. The AUV will capture the same scene during the detection/intervention stage, but most likely from different viewpoints. Afterwards, the system particularizes the well-known Perspective N Point problem (2D–3D) (Bujnak et al., 2011; Mei, 2012) for the specific application of scene recognition underwater. Results in the sea show the success in the use of this technique to recognize the scene where the object lies on the seabed, despite the presence of mud, turbid waters, sand, and different illumination conditions. It is the first time that all these techniques have been tested in different underwater environments, giving a solid and robust new method for operating underwater interventions.

This paper is structured as follows: Section 2 describes the hardware and software platforms of Fugu-f, Section 3 details the whole visual framework and functionalities, Section 4 presents experimental results obtained in diverse environments and finally, in Section 5 some conclusions and future work lines are drawn.

2. Platform description

2.1. Hardware

2.1.1. Overview

Fugu-f (Fugu-flexible) is mainly formed by two stereo rigs and a computer system linked through a firewire bus. This visual system was conceived as a flexible solution to provide visual perception to an AUV, as part of its sensor set or its payload. This system is a modularized solution that allows the use of different cameras and a variable geometry setting, depending on the mission to be carried out and the power and payload available on the host vehicle. Two examples of that modularity and flexibility recently tested are shown in Fig. 1, where Fugu-f appears aboard two different AUVs, the Girona500, a reconfigurable multipurpose AUV, designed by CIRS (Centre for Research in Underwater Robotics-University of Girona) (Ribas et al., 2012) (Fig. 1(b)) and Nessie VI from Heriot-Watt University (Maurelli et al., 2010) (Fig. 1(c)).

2.1.2. The hardware platform

Each module of Fugu-f is placed in an independent watertight case (see Fig. 1(a)). The Computer and the power supply unit are housed in a cylindrical enclosure made of anodized aluminum (dimensions: 19.5 cm ⊘ – 26.7 cm length) and each camera is enclosed in a sealed polyacetal resin case with a transparent frontal part made of methacrylate.

The cameras are connected to the computer cylinder through waterproof cables and connectors. Other connectors visible on the cylinder cover are used for powering and communication purposes. At present, the system is rated for up to 250 m depth.

The hardware includes (see Fig. 2)

- Two external Bumblebee stereo rigs: (1) a wide-angle camera with 97° of HFOV (Horizontal Field of View) in air, and a focal length of 2.5 mm, and (2) another camera with 66° of HFOV in air, and a focal length of 3.8 mm. Depending on the tasks to be carried out, the cameras can be mounted in different orientations. For example, in TRIDENT, for visual odometry, the wide-angle camera was used in a bottom-looking orientation so it could gather a wider field of view of the seabed and the overlapping area from frame to frame was larger. However, the camera with a narrower angle was more convenient in that position during the manipulation task, where precise localization of the target was needed. This is one of the main advantages of this system when being installed in different vehicles. The baseline of the camera is 12 cm. The depth resolution at 1 m with one pixel of disparity is 15.5 mm for the focal length of 2.5 mm and 10.7 mm for the focal length of 3.6 mm. These cameras have an IEEE-1394a firewire interface.
- A mini-ITX motherboard based on an Intel core with an i5 processor at 2.33 GHz, which holds the execution of all the operational software (cameras, image processing, sensor data reading and remote communications).
- A PCI express 2 port firewire card to connect the cameras to the motherboard.

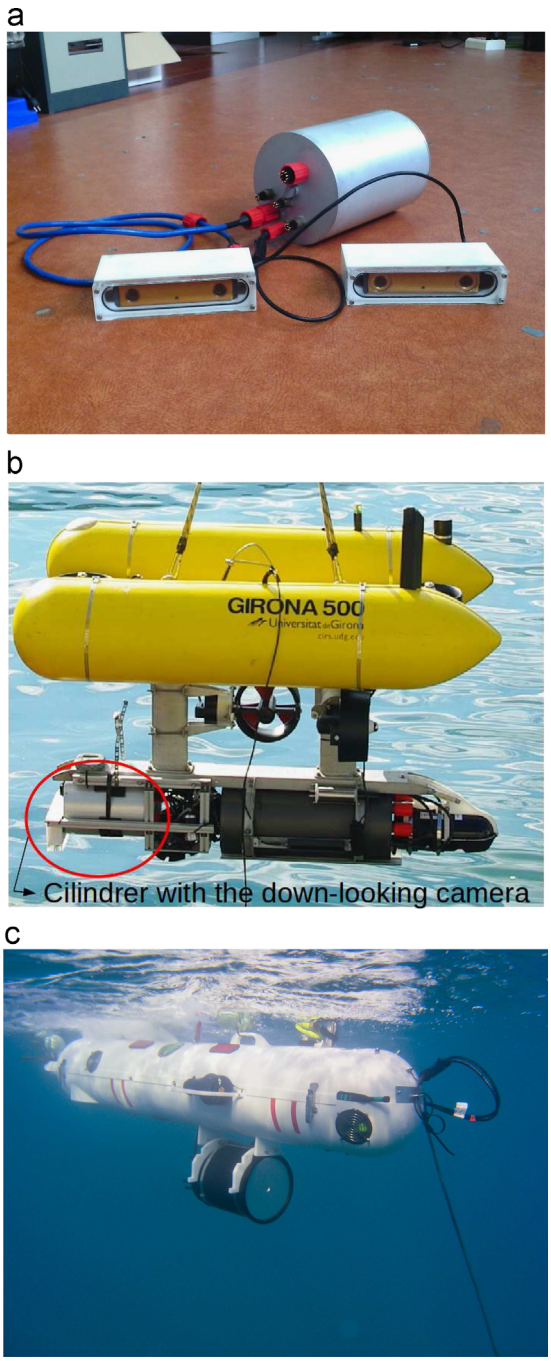

Fig. 1. (a) The Fugu-f cylinder with both stereo cameras. (b) The cylinder and one camera mounted on the Girona500 AUV. (c) The Nessie AUV with Fugu-f mounted in its payload.

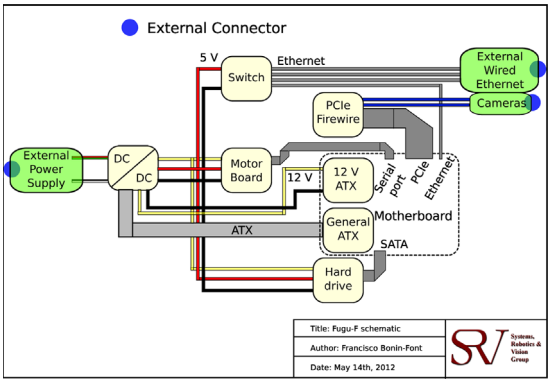

Fig. 2. The schematic of the Fugu-f hardware.

- A microcontroller-based board specifically developed for the project which manages some water leak detectors and a pressure sensor. It communicates with the computer via a serial port to manage the sensor readings.
- An ethernet Switch.
- An ATX 250 W DC–DC converter to supply power to the motherboard, the hard drive, the microcontroller board and the switch.

All the elements mentioned above except the cameras are enclosed in the cylinder. External connectors are placed in one lateral cover of the cylinder and include three connectors for ethernet communications which enter directly to the switch, another connector for external power supply and two more connectors for the cameras.

Another important operative parameters include

- Operating voltage: 8–30 V.
- Power consumption: 60 W (average) – 96 W (peak).
- Startup time: 6 min.
- Operating temperature: from −10 °C to 30 °C (water).
- Weight in air: 8 kg, while in water, the cylinder is neutral.
- Fugu-f can be installed in any vehicle with enough space in its payload. The maximum distance from the cylinder to the cameras is 1.5 m. Furthermore, the host vehicle needs to include an internal Ethernet connection in case it needs to communicate with the cylinder.
- The cylinder has free space available where a battery can be placed for the system to work in stand-alone mode.

Fugu-f can be powered from internal batteries, in case the vehicle has no external power supply, keeping a highly acceptable autonomy. Given an average power consumption of Fugu-f of approximately 60 W, a battery with a nominal voltage of 24 V and 10 A h would provide more than 4 h of autonomy.

Regarding the storage capacity of the system, any kind of standard device can be used (HD, SSD, SD card or USB drive). A selection criteria would only be based on the consumption and the storage capacity requirements.

2.1.3. Sensors

The pressure sensor used is able to operate from −40 °C to +125 °C, with a measuring range of 0–34 atm. The information given by this sensor is used for navigation and control.

Leak detectors are typically made of two simple wires placed near to one another, causing a detectable current drain when a drop of water puts them in contact. The data coming from this sensor is monitored continuously and its activation generates an alarm signal which is sent immediately to the computer of the host vehicle.

The stereo cameras are the main sensors of the system and they are used for many vision-based tasks that will be described in the following sections.

2.1.4. Communications

The visual module is equipped with the necessary elements for internal and external communications. It contains a switch that centralizes and distributes the communications with (a) other payload computers, (b) the central unit of the host robot, (c) remote PCs monitoring the system activity.

Fig. 3 shows two views of the Fugu-f hardware mounted on the cylinder internal structure.

2.2. Software

To reinforce software portability, reuse and sharing and to avoid crosscompiling, Linux (Ubuntu 10.04 lately updated to 12.04) was used as the operating system and all software modules were developed and run using the ROS (Robot Operating System) middleware (Quigley et al., 2009). ROS has a pure conception of distributed software facilitating its integration and interaction, improving its maintainability and sharing, and strengthening the data integrity of the whole system.

The software packages installed by default in Fugu-f can be classified into two categories: visual functions and internal state management function.

All the packages dealing with information provided by the stereo vision system are included in the visual functions category:

- Image grabbing and processing.
- Feature detection and tracking.
- Visual odometry.
- Stereo point cloud computation.
- Visual altimeter.
- Visual target identification and tracking.

An efficient management of potential dangerous situations that can affect the hardware integrity is crucial in order to preserve the physical system while it is operating underwater. Leaks, excessive temperature or even an energy drop off in the host vehicle can cause serious damage to the system. The internal state management function (the alarm module, in fact) comprises water leaks detection, temperature monitoring and power monitoring. When the alarm module detects a water leak, temperature excess or a

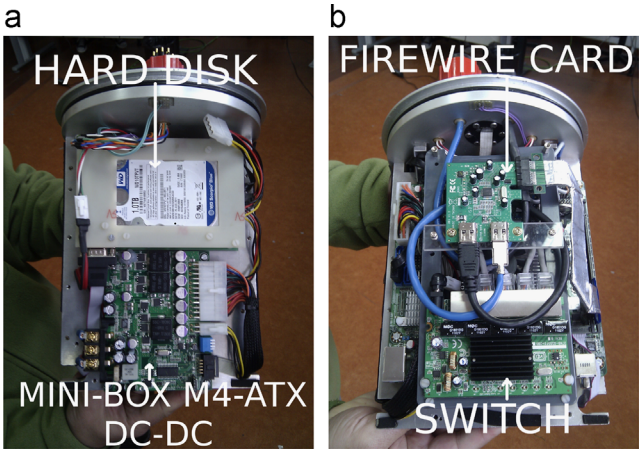

Fig. 3. Fugu-F: (a) The DC/DC converter and the hard disk. (b) The firewire card and the switch. The motherboard is visible under these cards.

powering drop off, it proceeds in 3 steps: (1) it publishes a ROS topic *Alarm* and its type, to alert the other computers connected to Fugu, (2) it terminates all running software, and (c) it shutdowns the computer. All software modules installed in Fugu-f publish their data as ROS topics. In this way, the information will be available to the rest of the systems present in the vehicle that can communicate with Fugu-f (in the case of TRIDENT, the free-floating manipulation module and any external laptop used to monitor the system activities). Likewise, Fugu-f is able to use all data published by the other systems connected to it.

3. Visual framework and control

This section overviews all current visual functionalities included in the Fugu-f system and listed above.

3.1. Image grabbing and processing

To save transmission bandwidth, raw images provided by the stereo camera are encoded using a Bayer filter and sent to the computer. The image processing is done afterwards in the computer side, including, in this order:

- A *debayer* interpolation to recover RGB and gray-scale stereo images.
- A rectification to compensate the stereo camera misalignment and the lens distortion. Two rectification processes are launched in parallel, one for the gray-scale images and another one for the color images. Rectified gray-scale images are used later for feature-based target identification, disparity and stereo odometry computation; color rectified images are used for 3D reconstruction, together with the disparity map. The intrinsic and extrinsic camera parameters for the rectification are obtained in a previous calibration process.
- Images are downscaled from the original resolution (1024×768) to a parametrizable lower rate which is strongly correlated with the available ethernet bandwidth; this down-sampling is done for monitoring the images from remote computers without saturating the ethernet communications.
- The computation of the disparity map from the monochrome images and the computation of point-clouds from the disparity map and the color images (left and right).

All this processing is done by several ROS nodes resident in the system. The resulting images and data are published as ROS topics, available to be used by other ROS modules, either internal or resident in other PCs connected to Fugu-f.

Fig. 4 shows a schema of the image processing tasks relation. The target identification and tracking modules are resident in Fugu-f while the navigation and the free floating manipulation modules reside in another computer. The description of the free floating manipulation module is beyond the scope of this paper (see for further details).

3.2. Feature detection/tracking and visual odometry

Robot motion estimation is essential in autonomous surveying or exploration tasks. Particularly, for the TRIDENT AUV, calculating accurately its pose is extremely important for its guidance and control, and for the data registering process.

One of the classical solutions adopted in autonomous systems equipped with cameras is visual odometry. Although visual odometry suffers from drift, it has been extensively discussed in the literature that it is useful enough in certain situations and presents less drift than a low cost or standard IMU.

The underwater domain presents several limitations and requirements to systems and vehicles that have to be taken into account in order to complete a functional approach. This media imposes 6 degrees of freedom to vehicles, thus having to take into account rotations in three axis and their possible singularities. Unmanned vehicles need an internal and limited power source, which means that computational performance has to be optimized. Visual odometry is usually based on tracking image salient points or features (Johnson et al., 2008; Fraundorfer and Scaramuzza, 2012). However, visual odometers based on feature tracking can easily fail, for example, with bad illumination conditions, in turbid waters, or over untextured sandy seabeds. The literature is lacking in visual odometry solutions efficient enough in real underwater environments.

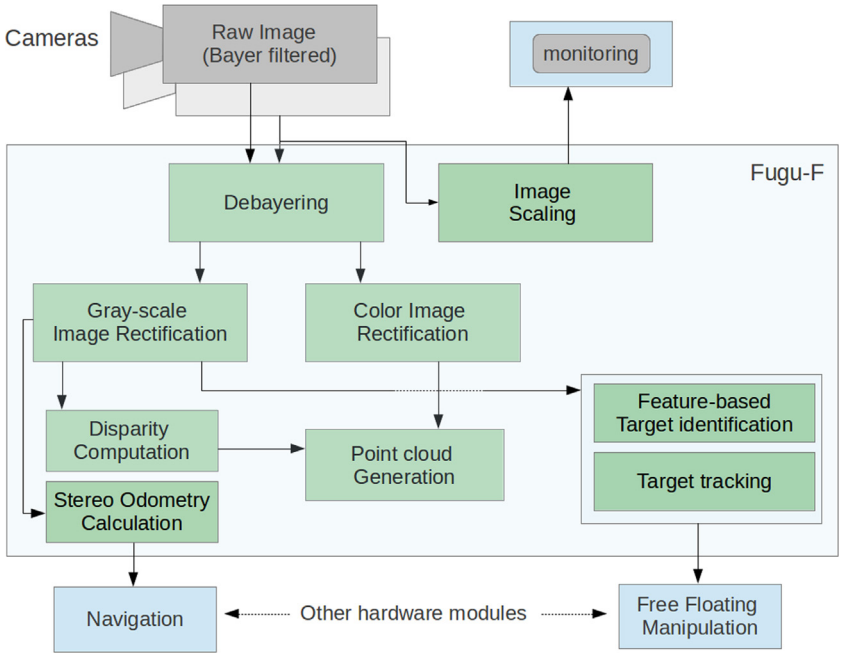

Fig. 4. The successive steps of the on-line image processing task.

Different terrains pose varied challenges to the odometers thus many empirical tests with underwater video sequences were needed in order to choose a visual odometer robust enough to be applied in both clear and turbid sea waters. Selecting and tuning properly the image feature detector and descriptor is crucial in underwater conditions to obtain accurate responses. According to the work of Garcia and Gracias (2011) Fast Hessian based detectors, and SURF in particular, demonstrates the best performance in terms of speed and accuracy when applied as feature detection and matching in underwater images.

If the camera is mounted on the vehicle in such a way that its lens axis is perpendicular to the ground, the vehicle moves only in surge, heave and yaw, and it is assumed that the ground is nearly plane, the motion estimation can be approximated in 2D through the homography (da Costa Botelho et al., 2009) that transforms image features inter-frames. Following this line, the initial implementation of our visual odometer used SURF features and the camera motion was estimated by computing the affine homography that transforms corresponding features between consecutive images. This assumption was, at the beginning, acceptable because (a) the vehicle navigated at sufficiently high distance from the bottom to consider the 3D structure of the terrain negligible, compared to the viewing distance, (b) the environments where the TRIDENT AUV had to operate were all along coastal areas with sand, seaweed or rocks and with no remarkable accidents on the terrain, and (c) the camera was mounted on the AUV with its axis perpendicular to the seabed.

Regardless of terrain, our experiments revealed an important issue: motion estimates based on images taken at high frequency but low resolution generated lower drift than those estimates based on low frequency but high resolution images. As a result, images were processed at the maximum speed of the frame grabber using the image resolution that could be processed at that speed.

However, the planar motion constraint over planar seabeds is too restrictive when using AUVs with 6DOF, if significant rocks or seaweed meadows are present in the environment and when the camera goes closer to the seabed for the intervention. In order to generalize the application of our system to different underwater environments, vehicles and applications, and taking advantage of the stereo cameras, a stereo visual odometer was finally adopted as being more suitable to cover a broader range of conditions.

Feature tracking-based stereo odometry has been applied in underwater robotics, providing good motion estimates (Corke et al., 2007), if the environmental conditions are reasonably acceptable.

To find a visual odometer with an acceptable undersea performance, we assessed and compared two recent approaches, LibViso2 (Library for Visual Odometry 2) (Geiger et al., 2011) and Fovis (Fast Odometry from Vision) (Huang et al., 2011). The assessment (Wirth et al., 2013) was done with Fugu-f installed on the Girona500 AUV, and in two different aquatic environments, first in a controlled one (a pool) and then in the sea. Experiments demonstrate that Viso2 outperforms Fovis in sea conditions, thus, the first odometer was finally chosen to run in Fugu-f during the TRIDENT sea trials (see Wirth et al., 2013 for further details).

Although the literature is profuse in visual odometers, these two approaches were initially chosen because of three main arguments:

- Both systems use simplified feature detectors and tracking processes to accelerate their overall response, so they are quite suitable for real-time stereo vision-based applications. Additionally, they have been tested in real platforms with high dynamics, Libviso2 in cars and Fovis in aerial vehicles.
- A pure stereo-3D process is used for motion estimation, facilitating its integration and/or reutilization with all the other software modules that implement 3D reconstruction and object recognition.
- The great number of feature matches makes it possible to deal with high resolution images; this point is very useful for stereo odometry.

3.3. Stereo 3D point cloud computation

When working in unexplored environments, 3D perception can strongly benefit some robot abilities, such as sensing and modeling the area surrounding the vehicle. The clear identification of objects is essential when they have to be manipulated. 3D models of natural sea-floor or underwater installations with strong relief are also very important sources of information for scientists and engineers. Visual photomosaics are a highly accurate tool for building those models with a high degree of reliability (Maki et al., 2008; Gracias et al., 2013). However, they imply a slow process and they are rarely applicable online. On the contrary, the concatenation of successive point clouds registered by accurate pose estimates permits building and watching the environment on-line and in 3D, but with a lower level of detail.

In TRIDENT, reconstructing the environment in 3D, and in real time, provided an additional knowledge of the area where the robot had to operate, facilitating the location of the target and the mission planning.

Thanks to the stereoscopy, the 3D coordinates of the scene points corresponding to image features identified simultaneously in each stereo image pair can be computed. These 3D points are commonly known as a point cloud. A point cloud is dynamic, changing as the robot moves and it can be more or less dense, depending on the amount of selected image points. Point clouds can be accumulated to build a dense 3D model of the environment. The real challenge comes in the 3D reconstruction pipeline from successive point clouds rather than the computation of an individual point cloud itself (Campos et al., 2011).

Real time object recognition and scene reconstruction are two of the principal tasks performed by Fugu-fusing stereo imaging. Fugu-fuses a classic algorithm for stereo multiple view reconstruction (Hartley and Zisserman, 2003): image points are matched in simultaneous stereo images; from these matchings, the disparity map is computed and the coordinates of the 3D points are calculated. Denser point clouds show reconstructions that are more realistic and reliable than other reconstructions done with sparser point clouds. In our case, all the pixels included in a certain configurable region of interest (ROI) are taken for the 3D point cloud computation. The concatenation and meshing of the partial reconstructed regions must be done according to the robot position, which should be as exact as possible to avoid map misalignments.

Due to the drift accumulated by a visual odometer, its use as a first estimation of the robot pose generates map distortions when employed on long routes. Even so, it is important to highlight that Fugu-f is able to build in 3D, at 10 fps and with a high degree of reliability, static objects or stereo sequences captured during relatively short time intervals. Fig. 5 shows, as an example, the 3D reconstruction of a car tire and its immediate surroundings, recorded during one of the sea trials described in Section 4.

However, on-going research includes the utilization of loop closing visual SLAM techniques to increase the accuracy of the robot pose estimation and thus the accuracy of the on-line 3D map reconstruction during longer trajectories.

3.4. The visual altimeter

Computing and controlling the altitude of the vehicle is crucial when the manipulator has to operate at a relatively short distance from the bottom. If the arm or the end-effector get too close to the

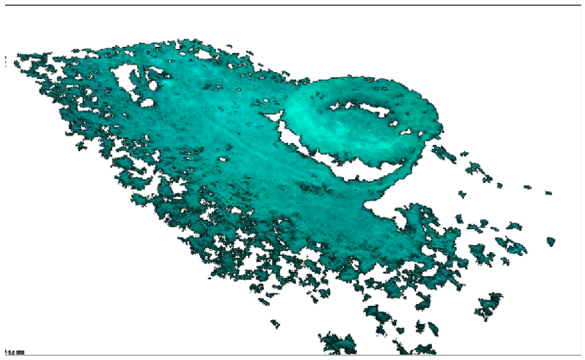

Fig. 5. The 3D reconstruction of a car tire.

see ground, they can hit it or be dragged along it, causing possible damages or miss-functions.

The visual altimeter receives as input a dense point cloud of a certain region of interest, captured from the bottom-looking camera. The module outputs the vehicle altitude as the median of the height (z coordinate) of all points included in that region.

The algorithm assumes that the portion of the environment included in the region of interest has slight height variations compared to the real altitude of the vehicle. Although this is not completely true in many real scenarios, it was demonstrated to be a very appropriate approximation to maintain the underwater robot at a constant distance to the seabed, during the navigation and manipulation experiments.

3.5. Visual target identification and tracking

3.5.1. Overview

The manipulation of an object requires its previous identification and tracking, which means recognizing the object to be manipulated and determining its relative position and orientation continuously.

The target tracking task must be synchronized with the vehicle control module, in order to keep the target approximately centered in the image and thus accessible to the manipulator.

Two different robust algorithms for target identification and tracking have been implemented and tested in TRIDENT: a color-based and a feature-based solution.

Four stages are needed in this supervised learning and pattern recognition procedure:

1. *Model definition*: Given an image and a mark for the target (e. g. an outline) the system has to generate the target abstract model for usage in the following phases. One of the novelties of the whole system is that there are no models taken or defined a priori, contrary to other approaches. As it is explained in the introduction, the mission consists of two stages. In the first stage, the submarine navigates surveying the environment where the intervention has to be done, recording video of all the explored area. Then, before the second stage starts, a human operator views those video sequences and identifies the object on the sea bottom just by visual inspection. Since, in the sea, the target to be recovered can be partially covered by algae or sand, surrounded by rocks, deformed or broken, the operator defines the model by selecting in the recorded images what he thinks is the target. Simultaneously, the operator establishes the vehicle and manipulator maneuvering sequences for the recovering process. In real sea conditions, the operator most likely will select the object and a part of its surroundings, and if the object is partially occluded by algae or sand, this will also be in the model.
2. *Target detection*: In the second stage, when the submarine operates autonomously to find and to recover the target, the visual system must identify the same scene selected by the operator, not just the object according to a model defined offline, but the object located in the sea bottom. The scene viewed by the robot during the second stage will be the same as that recorded during the survey, but probably captured from a different point of view. That is why it is so important for a successful object/scene recognition in many different sea conditions.
3. *Pose estimation*: Once the target has been detected in the image, its exact pose has to be estimated to determine the best way of approaching for the planned intervention.
4. *Tracking*: The relative position of the target in the image has to be permanently monitored. Water currents and changes in the dynamics of the vehicle-arm-hand system cause the target to move, so the proper orders have to be generated and sent to the free-floating manipulator to assure a successful grasp.

Some assumptions have to be made in order to formalize the target detection task:

- The camera plane is mostly parallel to the seabed (orthogonal view) and the distance to the ground is known. In this way, the possible motion can be limited to 2D (rotation and translation). The object descriptors used for detection have to be translation, rotation and scale invariant.
- The target and the background are static (before intervention), so the scene does not change significantly.
- The vehicle carries its own light-source. Light conditions are almost constant and the feature descriptors must be invariant to changes of brightness.
- The appearance of target and background cannot be trained in advance; online learning is needed.
- The target itself may not be homogeneous in color or texture.

3.5.2. Descriptors, detection and tracking

The success of the target identification tasks depends entirely on the way the object model is characterized.

The present problem of object detection can be reduced to a problem of foreground-background separation, similar to other standard pattern classification problems. Each pixel or region of the online image is classified either as foreground (i.e. as part of the object) or as background depending on the comparison of its descriptor with the object model descriptor.

Descriptors must accomplish certain general conditions to give accurate results in underwater applications:

- *Robustness*: The detection has to be invariant to translation, rotation, scale and brightness.
- *Speed*: The object detection/tracking module has to publish its data fast enough to permit a quick reaction on the navigation module, otherwise the target can go quickly out of sight.
- *Accuracy*: To minimize false detections or missing points, descriptors have to separate the target from the background precisely.

Color extraction methods for object detection have been used underwater since there are certain colors highly observable even if the visibility conditions are degraded (Yu et al., 2001). Color correction, restoration or filtering, together with techniques for image quality enhancement improve the application of color-based element detection techniques underwater (Schettini and Corchs, 2010; Lee et al., 2010).

The first tests with Fugu-f were done using the Histogram Back-projection algorithm (Swain and Ballard, 1990) but applied on the hue and the saturation channels from the HSV color-space.

Histograms in the HSV space are invariant to translation, rotation and illumination changes, and their computation time is minimal. First of all, the end-user must play the image sequence grabbed during the survey stage of the mission, and select the object model from one of the frames. Once the model has been stored, the object has to be identified and tracked during the autonomous detection and intervention stage.

The object tracking procedure has two steps: (a) the object in the current image is scaled to match its number of pixels with the number of pixels of the object model, (b) then, the detected object is rotated and translated until a maximum of overlapping with the object model is found; the degree of overlapping between the model and the detected target indicates the difference in scale between both. Once the target is detected, the coordinates of its bounding box are published to drive the manipulator during the grasping process.

Morphological operators were used to filter out camera artifacts and small false detections.

If the object model is selected in such a way that part of the background surrounding the object is included, the detector shows some false positives around the real object. Furthermore, the algorithm can also fail in those situations where the colors of the object and the background does not present consistent differences. For example, in underwater terrains where the object could be partially covered by sand or seaweed, its detection could be seriously compromised. To overcome these situations, a feature based detector was developed as a more robust alternative in certain cases.

If the target or its surrounding area are textured enough, then the object model can consist of a set of interest points (features) and their descriptors. In this case, the model will contain not only the features generated by the object itself, but also those features generated by the background surrounding the object. Then, the target is detected by using the features/descriptors of current images with those of the training model. Different training and observed data can be combined to perform this type of object detection. Table 2 summarizes the four possibilities explored in this project. Column *Training* refers to the nature of the features used to characterize the model, 2D for image features and Stereo for 3D points of the scene. The *Observation* column refers to the nature of the points computed online. Column *Output* refers to the nature of the output data.

The simplest approach consists of relating, via a homography, the 2D features extracted from the model image to the 2D features extracted from the online images. If the homography exists, the object is recognized although it could suffer a translation, and a rotation with respect to the model. This approach is restricted to coplanar points in the scene.

The third and the fourth approaches listed in Table 2 permit the 3D pose of the object to be recovered. Although these detectors were implemented, they could not be tested in the field experiments due to the difficulties in the definition of the 3D object model when the target was not clearly distinguished from the background.

Table 2
Different combinations for object recognition.

Training	Observation	Output
2D	2D	Homography
2D	Stereo	6DOF camera pose
Stereo	2D	6DOF object pose
Stereo	Stereo	6DOF object pose

Consequently, the second option (training in 2D- detection in 3D) turned out to be the one finally used in all the missions where Fugu-f participated. This solution permits the recovery of the camera pose during the training phase with respect to the system of coordinates attached to the current camera pose (stereo).

The algorithm is not restricted to any type of features or descriptors, but for real experiments the ORB features and descriptors (Rublee et al., 2011) were used because they demonstrated empirically the accomplishment of the general requirements of robustness, speed and accuracy.

This solution is a particularization of the Perspective N-Point (PNP) problem and refers to the process of computing the absolute camera pose given its intrinsic parameters and a set of 2D–3D point correspondences. This problem can be found in the literature formulated in numerous solutions and can be applied in a wide range of applications such as structure from motion, object recognition or even for localization (Bujnak et al., 2011; Mei, 2012).

This method has been adapted to our problem as follows: (a) 2D features (s_i) are extracted from a single training image (the left from the stereo pair, for example) and stored together with their descriptors; (b) correspondences between the current left image and the model are found by matching the 2D feature descriptors of both images, and the homography H that transforms those matchings from one image to the other is computed; this homography is valid assuming again a planar motion and a seabed with no significant relief, (c) matched features are back-projected from the current image to the 3D world using the stereo geometry, (d) these 3D points are re-projected onto the initial training image plane assuming the existence of a rotation and a translation:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} [R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{1}$$

where $(u, v, 1)$ are the homogeneous image coordinates of the re-projected point, $(X, Y, Z, 1)$ defines the 3D world point to be re-projected, (f_x, f_y) is the focal length, (c_x, c_y) is the principal point, and $[R|t]$ is the matrix that describes the camera motion around a static scene or vice versa.

The selected value of $[R|t]$ is the one that minimizes the total re-projection error:

$$[R|t] = \text{argmin}_{R,t} \sum_{i=1}^{N} \|p_i - s_i\|^2 \tag{2}$$

where p_i is the re-projected image point (u, v) and s_i is its corresponding original feature on the training image. Eq. (2) can be solved iteratively using Levenberg–Marquardt algorithm (LMA), also known as the damped least-squares (DLS) method (Pujol, 2007). Fig. 6 illustrates this theoretical approach.

$[R|t]$ turns out to be a transform *TF* that gives the position of the camera when the training image was taken with respect to the current camera pose. Obtaining a rotation and a translation from Eq. (2) with a minimum error means that both scenes are approximately the same, but taken from different points of view. In consequence, if $[R|t]$ exists, the scene captured as the model is recognized, otherwise, it is not.

Fig. 7 illustrates the idea of the transformation. WC is the north–east–down oriented world coordinate system, CP is the camera coordinate system with respect to WC, OC is the object coordinate system with respect to CP and Mod is the training frame. The existence of CP is perfectly valid since this is a stereo system and one can always calculate all 3D scene points with respect to the camera using the epipolar geometry. But in TF, there is no system of coordinates, a priori, since it is just an image in 2D. TF is the 3D transform that characterizes the pose of the camera

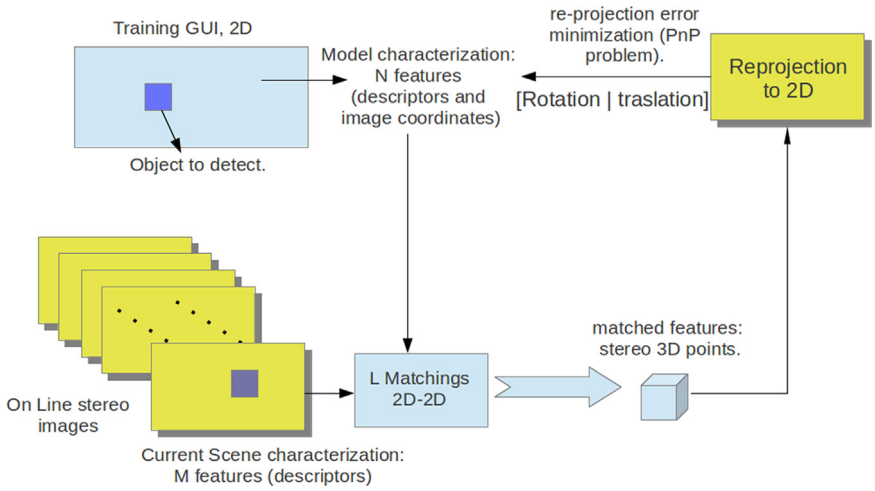

Fig. 6. Target detection and tracking using a feature matching-based method.

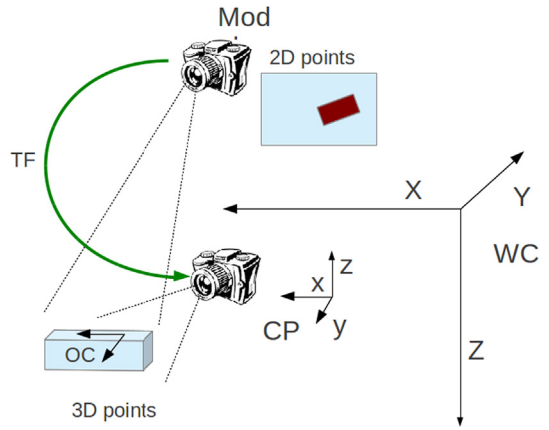

Fig. 7. Transform given by the resolution of the 2D–3D re-projection problem.

when the training image (Mod) was taken with respect to CP. By inverting TF, the transformation of the current camera pose to the pose that the camera had when the model was defined can be computed online. TF permits projecting all 3D points from CP to Mod, but not at the inverse, that is, from Mod to CP.

TF^{-1} is published as a ROS topic to be used for the vehicle or for the arm/end-effector controller in order to correct their motion and position according to the motion of the camera to have the object always inside its FOV.

Once the model scene has been identified, knowing the object position inside the image and the grasping points is also essential for the manipulation task. Thus in the training phase, the end-user selects on the model a set of points that form a polygon surrounding the object, and the origin and orientation of the object coordinates system. These points and axis do not define the model, the object model is defined by the entire image selected by the end-user, but they are important for the grasping task. During the online detection these points and the object coordinate system are transformed using the homography H, from the training model to the current left image. These transformed points and the orientation of the object coordinate system are also published as ROS topics to be used by the Free-Floating Manipulation module to plan the grasping task. The pose of the target points are referred to the camera coordinate system, but they are transformed to the end-effector coordinate system through the camera-arm relative pose and the arm kinematics (Prats et al., 2012).

4. Experimental results and discussion

4.1. Overview

A wide range of experiments have been carried out with Fugu-f in the context of the TRIDENT project. One of the objectives of a typical mission in TRIDENT is the detection and recovery of an aircraft black-box mock-up. Departing from a vessel or a central station, the autonomous vehicle has to survey the area of interest until the previously defined target is found. Once the target has been detected, the vehicle approaches it and stays just above it to facilitate the manipulation and grasping maneuver.

Consequently, for this particular project, the functionalities offered by Fugu-f became fundamental for the mission success.

Meetings and workshops were regularly organized during the TRIDENT project to integrate the work of every partner and to test the required global functionality on real AUVs, first in pool conditions and then in real subsea scenarios.

Numerous training trials with the Girona500 AUV were first done in the CIRS pool. In these sessions, the bottom of the pool was covered with a printed digital image of a coral reef, to simulate a marine-like context (see Fig. 8). In these conditions, the poster was also used to calculate the trajectory ground truth by comparing the features extracted from the images of the bottom captured by the cameras with those features extracted offline from the original printed image, and applying a re-projection and optimization process equal to the one described in the previous section.

The ground truth computation is very precise since outlier matchings are eliminated by RANSAC and the error in the re-projection and optimization process is minimized.

Computing the ground truth with other exteroceptive sensors, such as laser or sonar, was not suitable in this case: laser would be useless in water and sonars provide data with low resolution. Using external infrastructures, such as LBLs (Long Baseline Acoustic Positioning System), was out of the scope of the project. Obtaining the ground truth from images grabbed online was the only possible method in this particular application. The trajectory ground truth for the experiments conducted in the sea was computed using mosaicking techniques.

Regarding the experiments in sea conditions, two trials were organized to test the project progress. The first one took place in Roses bay (ES) in September 2011 and the second one in Sóller Port (ES) in October 2012 (see News and Events section in www.irs.uji.es/trident).

In all these experimental meetings, Fugu-f was installed in the payload area of the Girona500 AUV with diverse arms and end effectors to be tested. In different sessions, various modules were extensively tested: the image grabbing, the online video visualization, the localization/mapping algorithms, the target identification/tracking and the grasping-manipulation.

For all the experiments detailed next, the images were downsampled from 1024×768 to 512×384 pixels to reduce the execution time.

4.2. Navigation and localization

Images provided by Fugu-f during all the experiments conducted in Roses were used: (a) to compute the stereo odometry, and (b) to build a photo-mosaic of the sea bead according to the approach presented in Ferrer et al. (2007) and Prados et al. (2012). The mosaicking task was performed offline by the CIRS staff, and its global result is the mosaic itself and the set of camera poses.

Fig. 9 shows two images gathered during the mission in Roses, evidencing the deficient lighting conditions and the lack of texture on the ground, composed of mud and typical port debris.

An accurate composition of the set of camera poses leads to the robot trajectory that was later used as the ground truth for other visual navigation experiments. From now on, units are expressed in trajectories and depth ⇒ meters; orientations ⇒ radians.

Fig. 10 shows the photo-mosaic of a portion of the Roses harbor sea floor, computed from the images gathered by Fugu-f during one of the surveys at a constant depth. The tire shown in Fig. 5 can be observed in the central area of the mosaic. Fig. 11(a) and (b) shows the robot trajectory computed by the mosaic algorithm, and Fig. 11(c) and (d) shows the trajectory resulting from the visual odometry computed with LibViso2.

Fig. 12 shows the vehicle orientation in roll, pitch and yaw, along this trajectory. Plot 12(a) corresponds to the set of vehicle orientations computed by the photo-mosaic algorithm and plot 12(b) shows the orientations calculated by the stereo visual odometer. Both algorithms returned the orientations in the quaternion space, but these values were transformed into the Euler space for easier representation and understanding. The most significant variations of the vehicle orientation are in its heading. As a matter of fact, Girona500 did not rotate significantly in roll or pitch, it only moved in surge, heave and yaw. See in plot 12(a) the important variations on yaw, which correspond to the changes of the vehicle heading along the survey trajectory. See also the evolution of the roll and pitch values close to 0. However, pitch and roll computed by the visual odometer present maxima close to 1 radian, causing an evident drift of the trajectory in the z-axis. The yaw angle of the odometry also presents errors with respect to the ground truth. Furthermore, maxima in yaw are also much higher than those computed from the mosaic data. Errors in the visual odometry are mostly due to the lack of reliable features in the roses seabed and the deficient illumination conditions (see Fig. 9).

Fig. 13 shows the trajectory of the Girona500 navigating in the CIRS pool with Fugu-f mounted on it. In this experiment the vehicle navigated at approximately constant depth and moved only in surge, heave and yaw. Plots 13(a) and (b) show the trajectory in 3D and 2D, respectively, corresponding to the ground truth and the visual odometry.

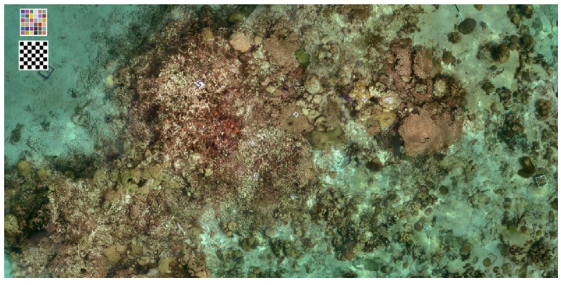

Fig. 8. The poster for the experiments on the CIRS pool.

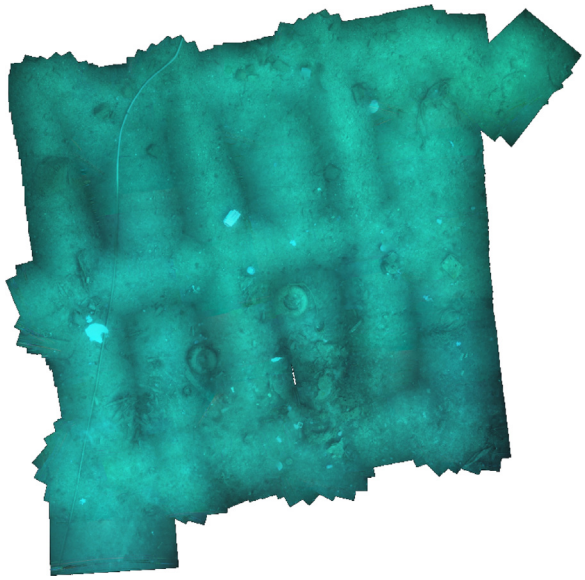

Fig. 10. A raw photo-mosaic of a portion of the Roses harbor sea floor. Mosaic courtesy of CIRS (see Gracias et al., 2013 for further details).

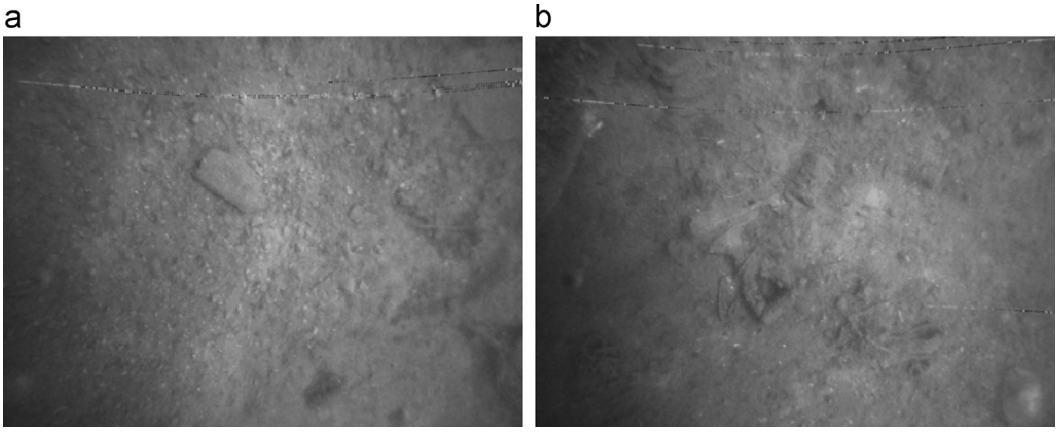

Fig. 9. Two images taken online during the Roses Trials.

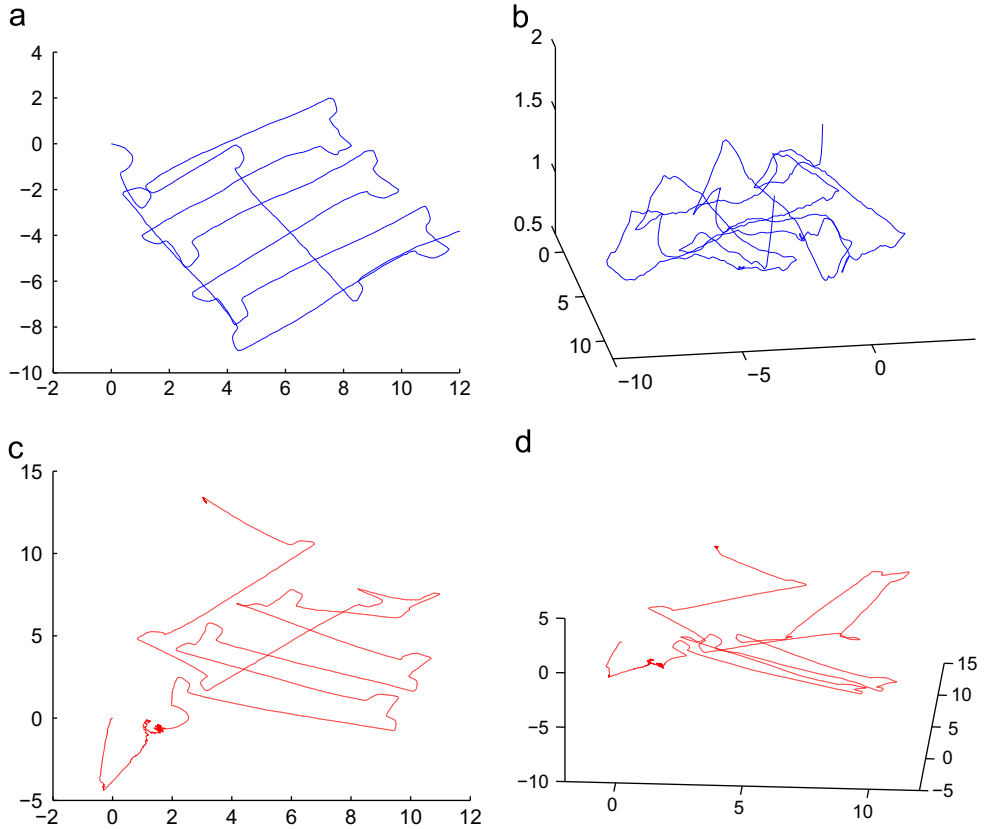

Fig. 11. Roses trials. (a) Mosaic trajectory in plane (x,y). (b) Mosaic trajectory in 3D. (c) Odometry in plane (x,y). (d) Odometry in 3D. All units are expressed in meters.

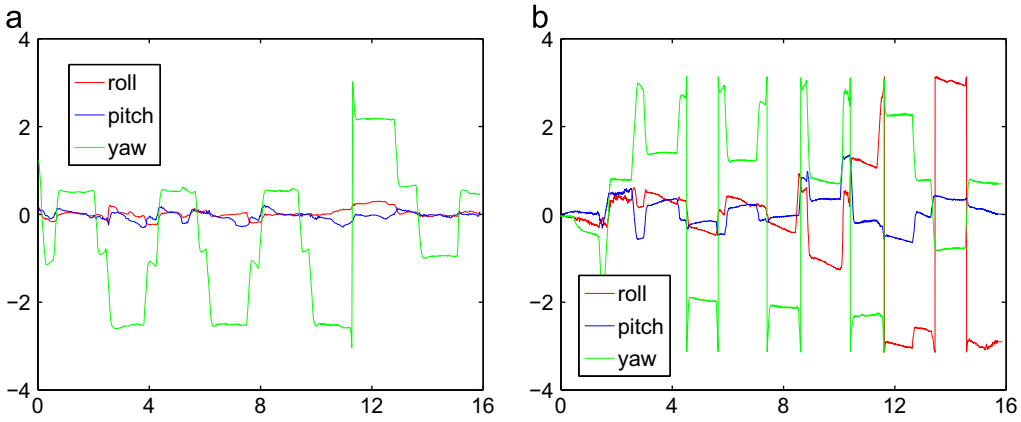

Fig. 12. Roses trials. Vehicle orientation at each point of the trajectory plotted in Fig. 11. (a) Mosaic ground truth. (b) Visual odometry. Horizontal units correspond to time expressed in minutes. Vertical units are expressed in radians.

Fig. 14 shows the evolution of the vehicle attitude expressed in Euler angles. Important variations in yaw reflect the changes on the vehicle heading when it passed from π to $-\pi$.

The small variations in roll and pitch reflect the slight motion of the vehicle in those angles plus certain drift.

The mean average translation error for CIRS Trial 1 is 0.010325 m, and the mean yaw error is 0.003347 rad, both computed according to Geiger et al.

The trajectory in the x–y plane is close to the ground truth, but like in the trial of Roses, it presents an important drift in z. This drift in z is mostly due to the error accumulated in pitch, reflected in Figs. 14(b) and 12(b), and it can be compensated by using the data coming from the pressure sensor.

Fig. 15(a) presents the evolution of the vehicle depth according to the pressure sensor and the ground truth while Fig. 15(b) shows the depth according to the visual odometry estimates. For the sake of an easy comparison, all data have been referenced to 0. If the origin of the global system of coordinates is located on the water surface and set as north(x)–east(y)–down(z), the evolution of the pressure sensor data can be compared with the z coordinate given by the odometry and by the ground truth.

See how the pressure sensor and the ground truth data are bounded around 0 (as expected for a plane motion), while the z coordinate of the odometry biases in z.

In conclusion, results of these trials together with results presented in Wirth et al. (2013) show that LibViso2 has a

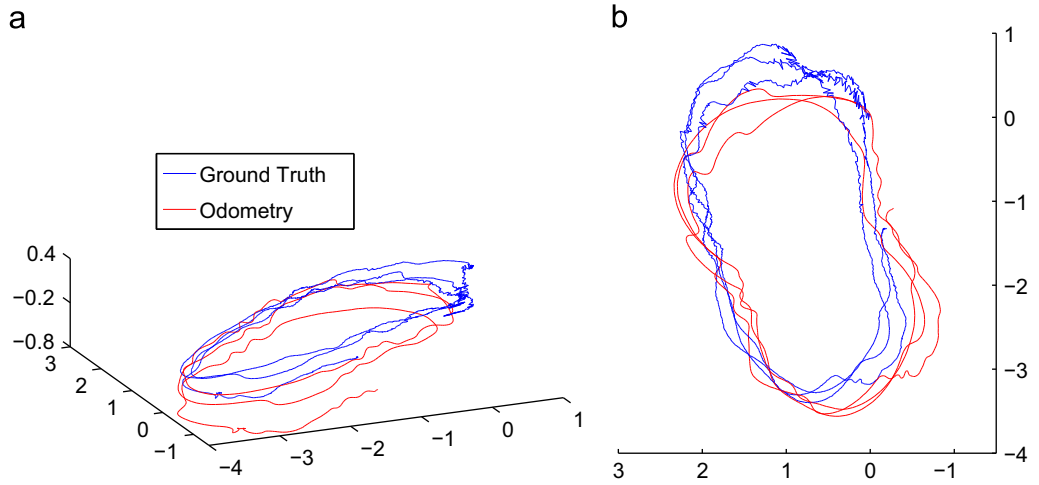

Fig. 13. CIRS trial 1. (a) Robot trajectory in 3D. (b) Robot trajectory in the x–y plane. All units are expressed in meters.

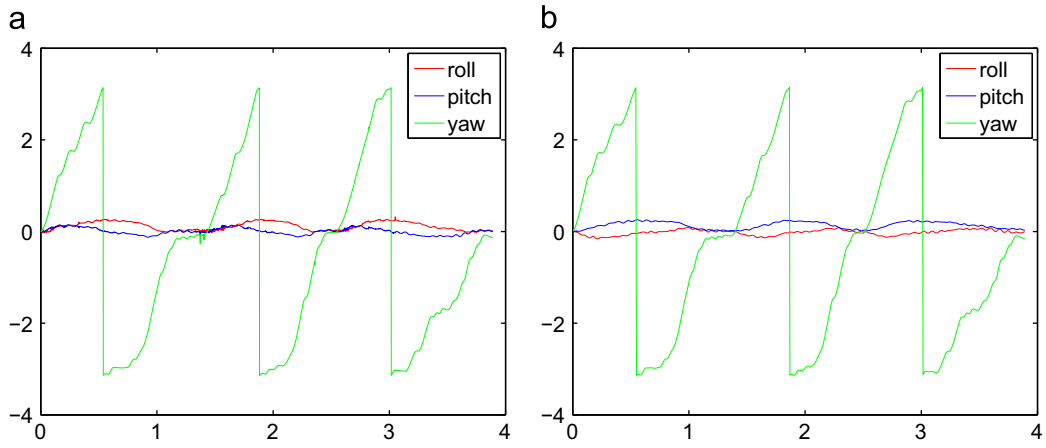

Fig. 14. CIRS trial 1. Vehicle attitude according to (a) the ground truth and (b) the visual odometry. Units on the horizontal axis represent time in minutes and units on the vertical axis measure radians.

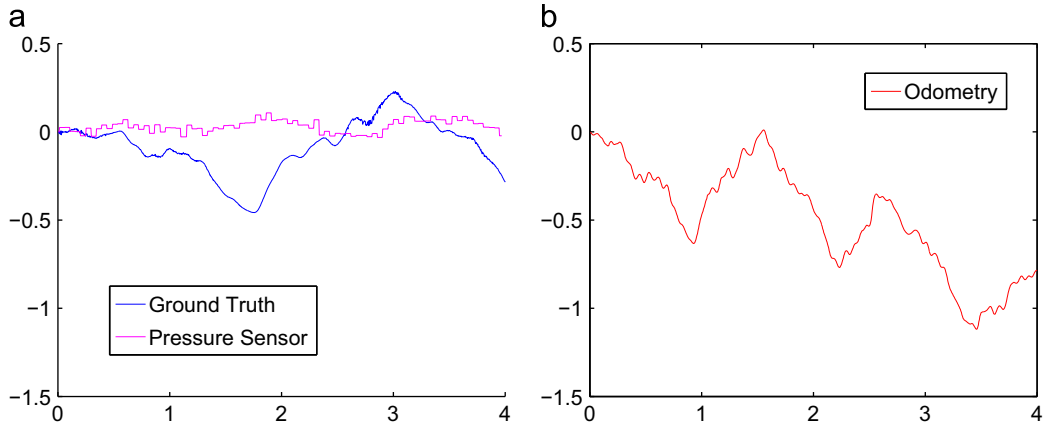

Fig. 15. CIRS Trial 1. (a) Depth estimated by the ground truth and the pressure sensor. (b) Depth estimated by the visual odometer. Units on the horizontal axis represent time in minutes and units on the vertical axis measure meters.

considerably good performance (especially in orientation) in underwater scenarios with relative good illumination conditions and sufficiently textured bottoms. When tested in real undersea scenarios, Libviso2 showed a non-negligible drift in pitch and yaw, but an acceptable performance in translation.

As it seems unavoidable getting drifts in pitch, roll and z, the use of certain navigation filters to integrate all data coming from all sensors (IMU, visual odometer, DVL, pressure sensor) is highly recommendable. These filters attempt to compensate drifts and to smooth the navigation data (Bonin-Font et al., 2013).

4.3. 3D reconstruction

Experiments carried out in Sóller (Mallorca) during the 2nd Field Training Workshop in Underwater Robotics Intervention included surveying, target detection and target grasping. For these

experiments Fugu-f was also mounted on the Girona500 AUV. Some of the surveying missions permitted the collection of images and pose estimations to build offline 3D views of the seabed.

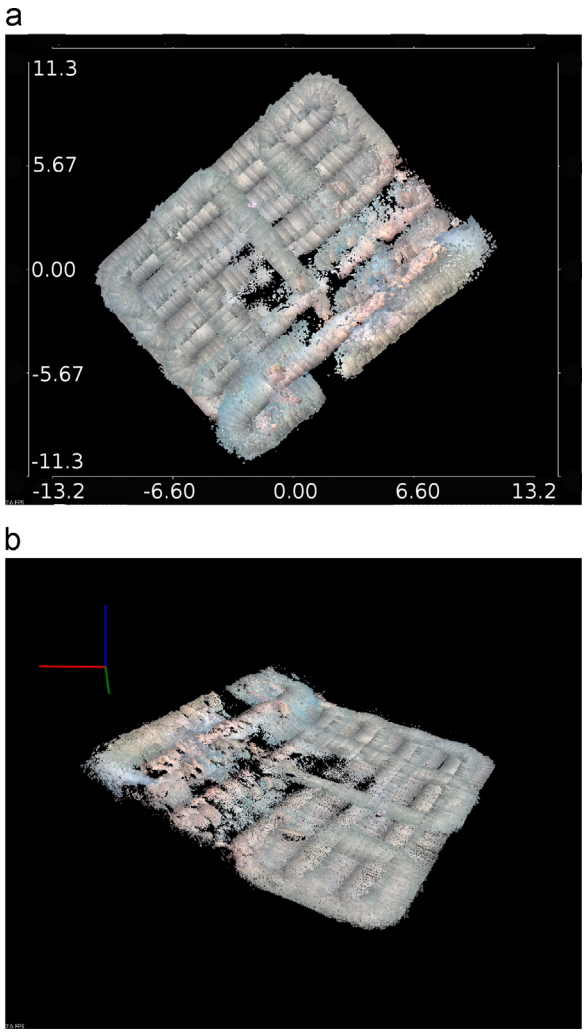

Fig. 16. Sea bed 3D reconstruction using point clouds and navigation data. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

The point clouds computed from the stereo views were assembled accordingly with the position estimates given by the Girona500 navigation module based on a DVL, a compass and a pressure sensor filtered with a standard EKF. These navigation data were used in order to minimize map misalignments due to biases in the visual odometry.

The dense clouds from the stereo views could contain up to 800.000 points from one shot. The merging of the successive point clouds in a map required the storage of huge amounts of data. In order to save processing time and memory resources, in that case, the rate of the processed stereo pairs was reduced to 1 per second and the data was down-sampled to only one point per voxel, heuristically determined at 5 cm^3. These results were particularly useful to get a general overview in 3D of the underwater scene where the robot had to operate.

Fig. 16 shows two views in 3D of part of the Sóller harbor where real experiments were conducted. Holes in the 3D maps correspond to zones where 3D points could not be triangulated due to bad illumination or blur. Fig. 16(a) shows a top-view (x–y plane) with the scale in meters, and Fig. 16(b) shows a view in perspective with the x–y–z axis plotted in red, blue and green. The axis has been plotted to give an idea of the absolute position of the 3D points with respect to the world coordinate system.

4.4. Target identification and tracking

The final objective of the experiments run in TRIDENT was the detection and recovery of an aircraft black-box mock-up.

To this end, the target detection and tracking algorithm resident in the Fugu-f was used.

Experiments in controlled and real scenarios demonstrated the efficacy of the object recognition and tracking algorithms, accomplishing the requirements of robustness, speed and accuracy.

Fig. 17 shows how objects that have a dominant representative color significantly different from the background colors can be easily detected. The selected model is shown in Fig. 17(a) and the detected target on the online video sequence appears in Fig. 17(b)–(d). All images show at the bottom the corresponding 2D H-S histogram.

The algorithm showed a successful detection in different positions and orientations of the black box, even with partial occlusion, running online with the video sequence (10 fps).

The real TRIDENT challenge was the application of the whole system in real underwater environments. Fig. 18(a)–(f) shows six frames of two different sequences grabbed during the Sóller trials.

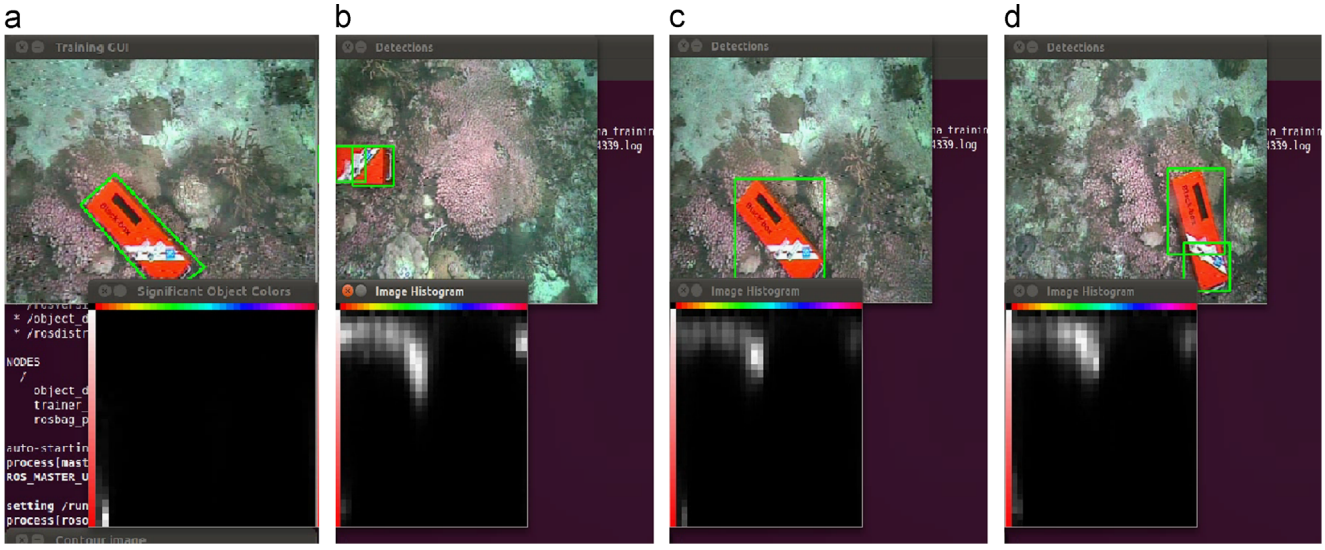

Fig. 17. (a) The selected object model and its H-S histogram, (b–d) the detection result on the online sequence with its corresponding H-S histogram.

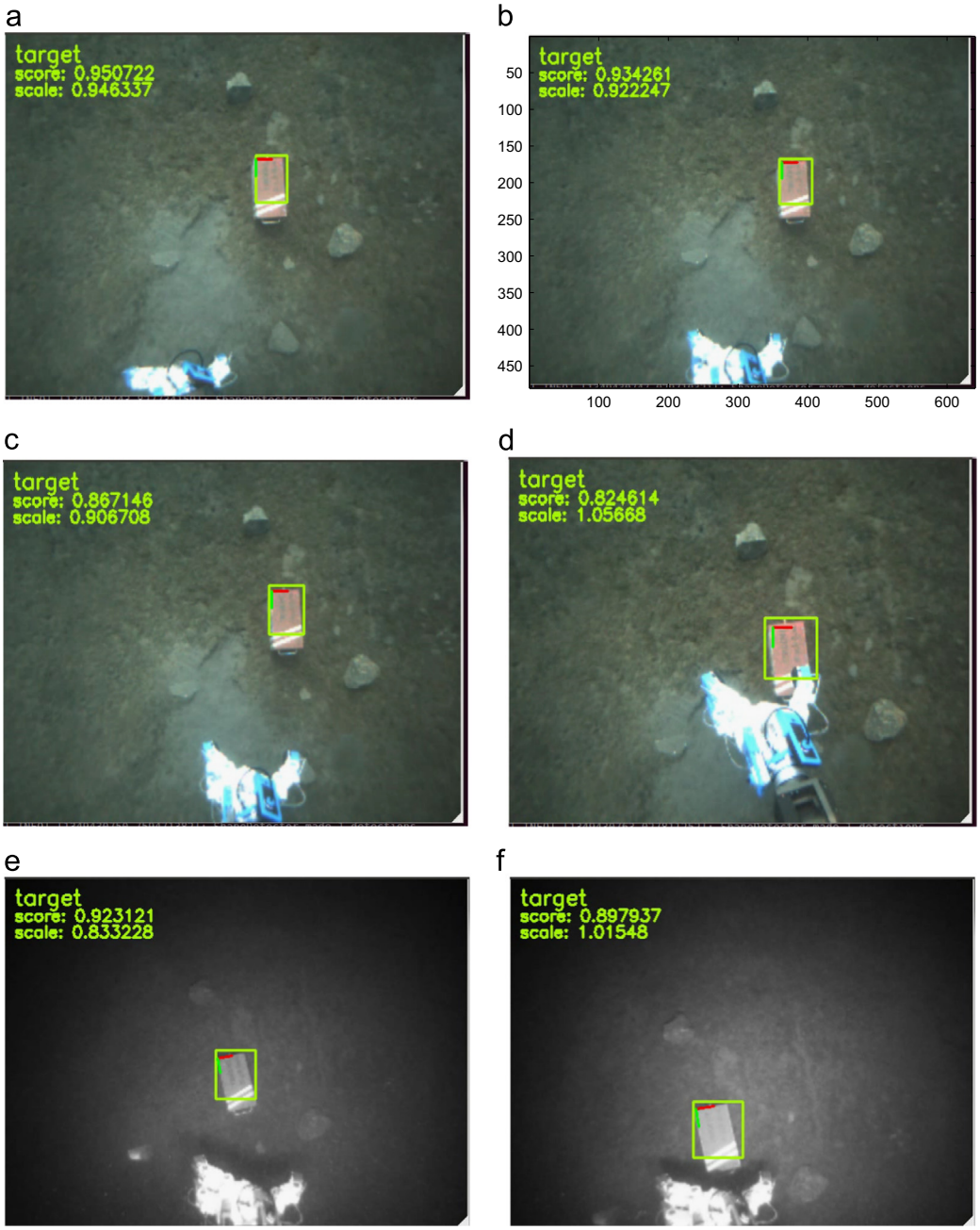

Fig. 18. Target identification/tracking and intervention in the sea using a color-based object detector. (a–d) During the day. (e, f) During the night.

Images from 18(a)–(d) were taken from an experiment done during the day and images 18(e) and (f) correspond to an experiment done during the night using the AUV led spotlights. Images show the detection and tracking of the black-box.

In both experiments, the robot was located over the target, with the lens axis of the narrow-angle camera perpendicular to the seabed. The end-user selected the model on the image and afterwards, the system was able to identify and track the object automatically. Using these tracking data, the control unit was able to generate the proper motion orders to the vehicle to keep the target continuously inside the FOV of the camera. Then, the robot started to move down to initiate the grasping process.

The lack of features in the muddy and untextured seabed, the blur and the deficient illumination conditions, especially at night, forced us to use the color-based algorithm in both experiments. The tracker worked without interruption as the robot moved down, which demonstrates the suitability of this algorithm for this particular environment. The parameter *score* shown in the images denotes the degree of identification with respect to the model and the parameter scale denotes its size with respect to the original model.

Illumination conditions can be improved by using external light sources, as a matter of fact, and as mentioned above, all the nightly experiments for TRIDENT with the AUV Girona 500 were done using the led spotlights of the vehicle. Alternatively, both current stereo cameras of Fugu-f could be changed by two cameras that integrate leds, with a slight increase in the power consumption of the cylinder and additional software for the lights management.

Figs. 19 and 20 show four frames of a video sequence corresponding to different tests carried out in the CIRS pool. Two different objects are shown, the black box (Fig. 19) and an amphora (Fig. 20). These tests are examples of the feature-based object detection algorithm. Images show successful detections when the object is scaled, translated, rotated and even partially

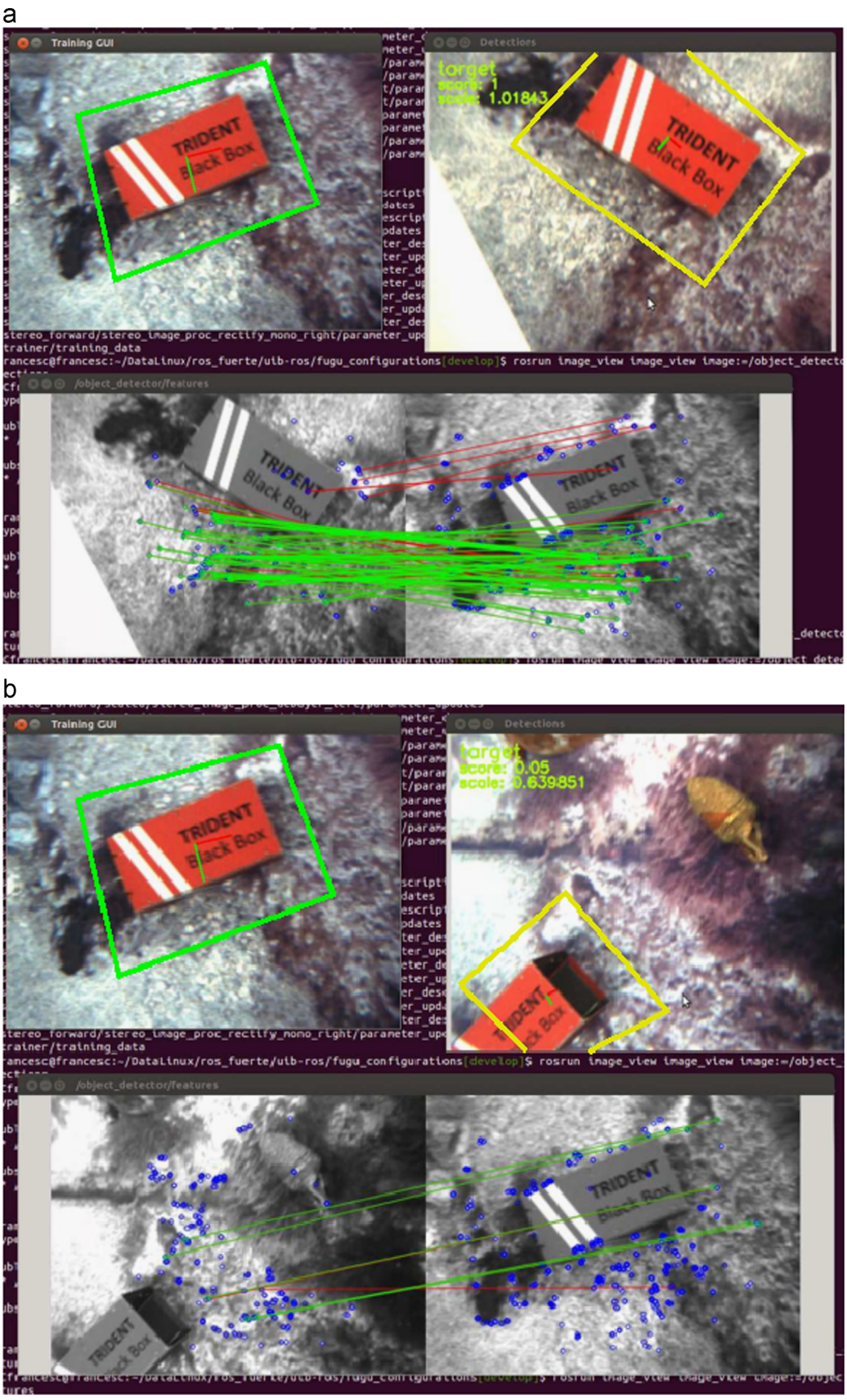

Fig. 19. Feature-based object detection. A test in the pool. (a, b) Detection of the black box. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

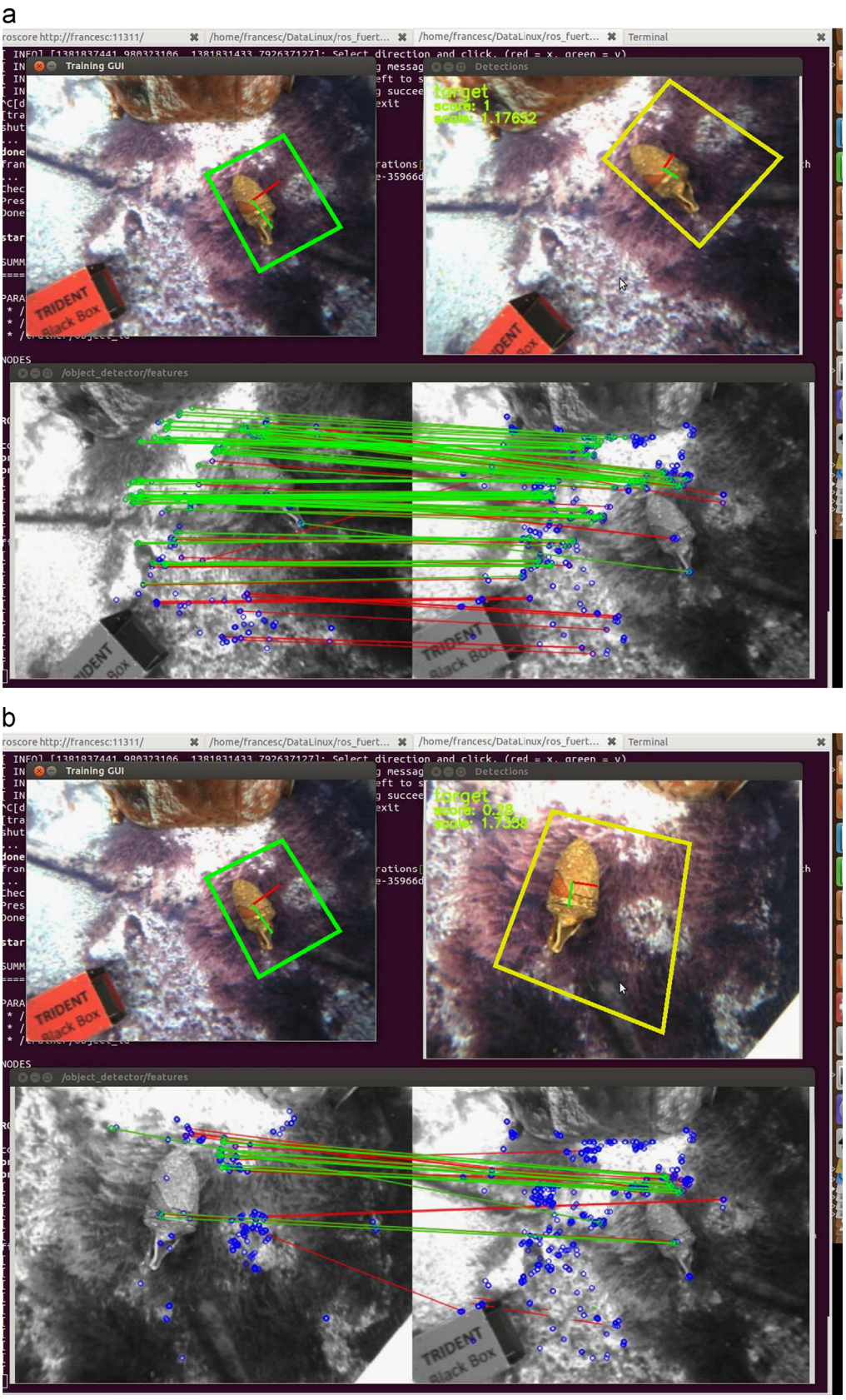

Fig. 20. Feature-based object detection. Another test in the pool. (a, b) Detection of the amphora. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

occluded with respect to the model (see Fig. 19(b)). The upper-left corner of every image shows the pop-up corresponding to the model, with the object surrounded by the polygon (green) and the *x*–*y*-axis of the coordinate system attached to it. The upper-right corner shows the object identified on-line with the transformed polygon and the object coordinates system with respect to the

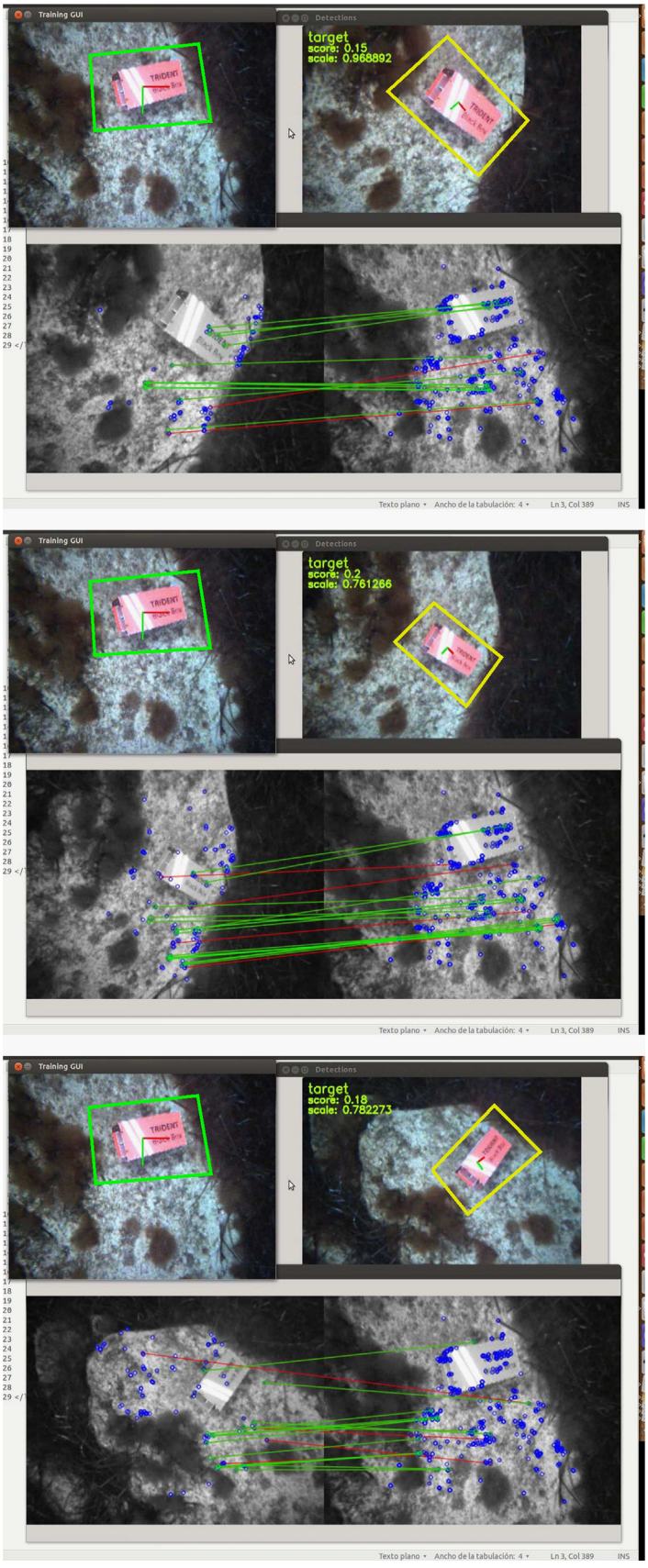

Fig. 21. Feature-based object detection. A test in the sea. Detection of the black box at different scales and rotations.

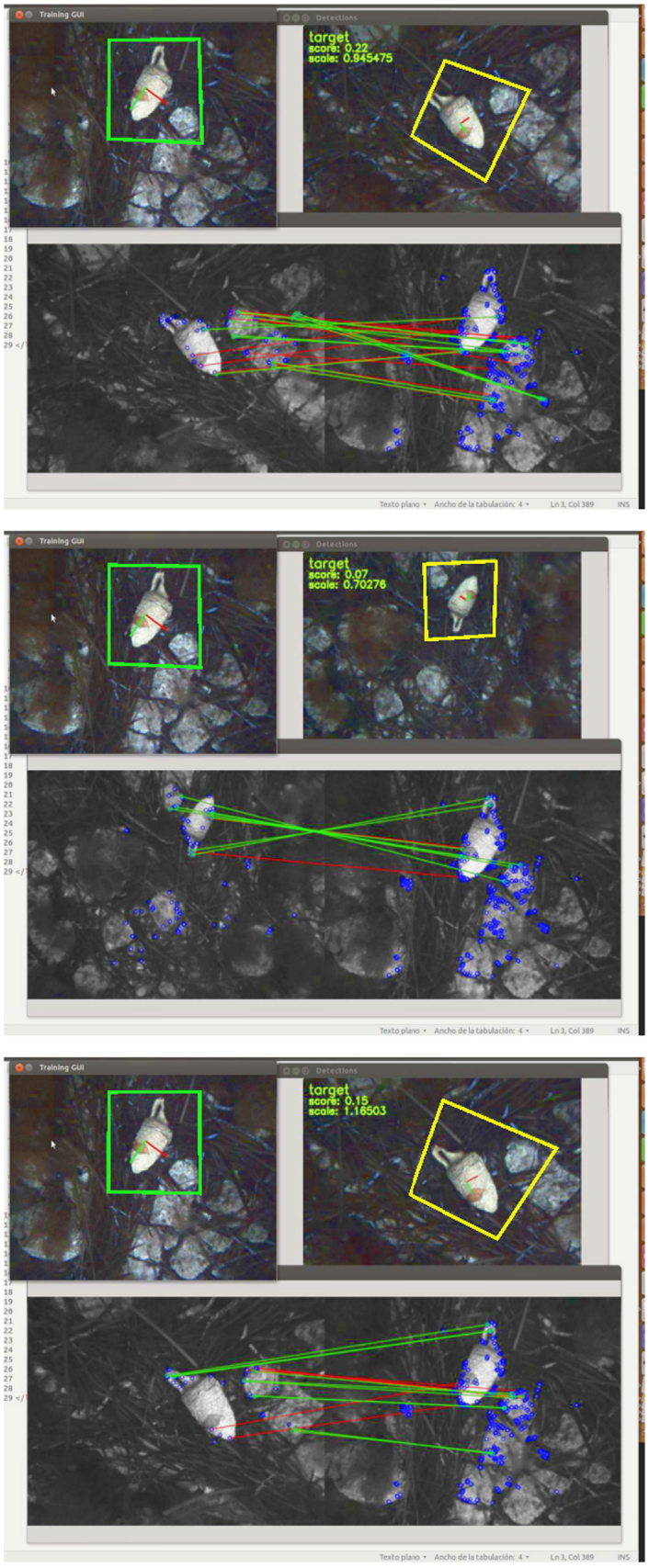

Fig. 22. Feature-based object detection. Another test in the sea. Detection of the amphora at different scales and rotations.

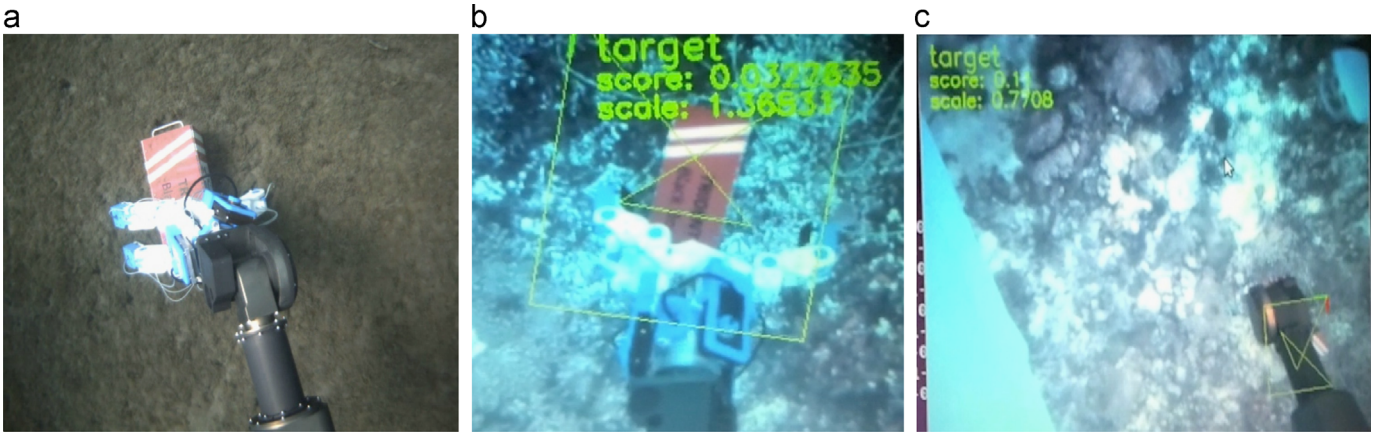

Fig. 23. The object to be manipulated (the black box) is partially occluded by the end-effector. (a) A representative case during a grasping maneuver in the sea. (b, c) Another representative case when running the grasping task in the CIRS pool, using the feature-based object detection algorithm. The detection is positive although the object is partially (b) or completely (c) occluded.

camera. Finally, the bottom frames show the feature correspondences between the model (right) and the current image (left). Features are plotted in blue and all the matches are depicted, but only the inliers have been colored in green.

Figs. 21 and 22 show six frames extracted from two video sequences grabbed during two experiments in the sea. Fig. 21 shows how the proposed feature-based algorithm detects the black box which is lying in a sea bed with rocks and algae. The detection is positive in different rotations of the camera and viewing the object from different altitudes. Fig. 22 shows the detection of an amphora which is lying on a seabed partially covered by posidonia oceanica. In both cases, the score, which represents the number of inliers, is low due to the texture of the seabed, especially with the posidonia, but still sufficient to detect the model.

This feature-based solution is especially relevant when the target is being partially occluded by the manipulator, as shown in Fig. 23. Performing a robust target tracking during the whole intervention process, regardless the position of the end-effector in the image, is essential to guarantee a stable and a precise manipulation maneuver.

It is important to remember that the model and object identification relies not only on the object features, but also on the entire scene where the object is placed. The target is taken as a part of the environment. Therefore, this procedure has several advantages compared to the color-based procedure: (a) if the model image is completely or partially contained in the online image, the object is detected, (b) if the object is partially occluded by algae, sand or any other elements, it will also be detected, (c) if the object lacks significant features, or the illumination conditions are not sufficiently favorable to detect all of them, the object is likewise detected thanks to the features detected in its surroundings. See, for example, in Figs. 20 and 22, how the amphora is detected even though this object does not generate any significant inlier.

In summary, the computation of the camera displacement with respect to an initial state and the motion in the image of several object points permit the vehicle or the manipulator to locate themselves adequately for an intervention as precise as possible.

The feature-based detection procedure is feasible mostly in scenes with acceptable illumination and texture, such as, for example, coastal or shallow waters with stones and seaweed mixed with sand. On the other hand, the color-based approach has shown to be effective in scenarios with deficient illumination conditions or with an untextured seabed.

5. Conclusions

Exploitation of subsea installations by offshore industry or underwater biology activities can make the most of robust and reliable visual platforms specially designed for marine environments. Exploration and manipulation capacities can clearly increase their precision by using vision based systems. This paper presents a new infrastructure, called Fugu-f, for underwater imaging in the form of an adaptable hardware module designed to provide visual abilities to host underwater robots.

The descriptions of the physical structure, the hardware, the software architecture and all the functionalities provided by this platform are included in this paper.

Experimental results carried out in water tanks and real sea scenarios have demonstrated the usefulness of the system in tasks such as surveying, navigation, localization, 3D reconstruction of the sea bed, mosaicking, object identification/tracking, and manipulation. The main advantages of this system with respect to other subaquatic visual infrastructures are (a) this is an external module, applicable to all kinds of vehicles that have room for payload, and external connections for power supplying and ethernet, avoiding changes on the vehicle structure, (b) it is flexible in its configuration, in hardware and in software; cameras can be placed in different positions depending on the mission to be carried out, and the installed software can also be adapted to the needs of the mobile platform, (c) it is truly robust: it has been protected against corrosion caused by salt, its watertightness has been assured up to 100 m depth and its internal hardware is completely isolated from interferences coming from other hardware and from static/dynamic spurious marine currents or voltages.

The system is able to reconstruct the environment, in real-time, with an important degree of reliability and, furthermore, it is perfectly able to detect and track objects that are previously modeled, in different scenarios with different environmental conditions.

Future work includes the use of 3D SLAM to reduce misalignments in the world reconstruction process over large areas, and the inclusion of optimization techniques to improve the reconstruction of small areas or static objects.

Acknowledgments

The authors are grateful to all the members of the TRIDENT consortium and especially to the CIRS (University of Girona) to put their facilities at our disposal. The authors are also grateful to the

Ocean Systems Lab at Heriot Watt University for contributing with the Nessie AUV to the first experiments with Fugu-f in real undersea scenarios.

References

Bagley, P.M., Smith, K.L., Bett, B., Riede, I.G., Rowe, G., Clarke, J., Walls, A., 2007. Deep-ocean environmental long-term observatory system (DELOS): long-term (25 year) monitoring of the deep-ocean animal community in the vicinity of offshore hydrocarbon operations. In: Proceedings of MTS/IEEE OCEANS European conference, Aberdeen, UK.

Bonin-Font, F., Burguera, A., Oliver, G., 2011. Imaging systems for advanced underwater vehicles. J. Marit. Res. 8 (1), 65–86.

Bonin-Font, F., Beltran, J.P. , Oliver, G., 2013. Multisensor aided inertial navigation in 6DOF AUVs using a multiplicative error state kalman filter. In: Procedings of MTS/IEEE OCEANS European Conference, Bergen, NO.

Bujnak, M., Kukelova, S., Pajdla, T., 2011. New efficient solution to the absolute pose problem for camera with unknown focal length and radial distortion. In: Lecture Notes in Computer Science, vol. 6492, pp. 11–24.

Campos, R., Garcia, R., Nicosevici, T., 2011. Surface reconstruction methods for the recovery of 3d models from underwater interest areas. In: Proceedings of MTS/IEEE OCEANS European Conference, Santander, ES.

Corke, P., Detweiler, C., Dunbabin, M., Hamilton, M., Rus, D., Vasilescu, I., 2007. Experiments with underwater robot localization and tracking. In: Proceedings of IEEE ICRA, pp. 4556–4561.

da Costa Botelho, S.S., Drews, P., Oliveira, G.L., da Silva Figueiredo, M., 2009. Visual odometry and mapping for underwater autonomous vehicles. In: Robotics Symposium (LARS), 6th Latin American, pp. 1–6.

Dunbabin, M.F., Roberts, J., Kane, U., Winstanley, G., Corke, P., 2005. A hybrid AUV design for shallow water reef navigation. In: Proceedings of IEEE ICRA, pp. 2105–2110.

Evans, J., Redmond, P., Plakas, C., Hamilton, K., Lane, D., 2003. Autonomous docking for intervention-AUVs using sonar and video-based real-time 3D pose estimation. In: Proceedings of MTS/IEEE OCEANS Conference, vol. 4, Sant Diego, USA, pp. 2201–2210.

Ferrer, J., Elibol, A., Delaunoy, O., Gracias, N., Garcia, R., 2007. Large-Area photo-mosaics using global alignment and navigation data. In: Proceedings of MTS/IEEE OCEANS conference, Vancouver, CA.

Fraundorfer, F., Scaramuzza, D., 2012. Visual odometry. Part II: matching, robustness, optimization and applications. IEEE Robot. Autom. Mag. 19 (2), 78–90.

Garcia, R., Gracias, N., 2011. Detection of interest points in turbid underwater images. In: Proceedings of IEEE European Oceans, Santander, ES.

Geiger, A., Ziegler, J., Stiller, C., 2011. StereoScan: Dense 3D reconstruction in real-time. In: Proceedings of IEEE Intelligent Vehicles Symposium.

Geiger A., Lenz P., Stiller C., Urtasun R. The KITTI Vision Benchmark Suite. Accessed 15 november 2014 on ⟨http://www.cvlibs.net/datasets/kitti/eval_odometry.php⟩.

Gracias, N., Ridao, P., Garcia, R., Escartin, J., LHour, M., Cibecchini, F., Campos, R., Carreras, M., Ribas, D., Palomeras, N., Magi, L., Palomer, A., Nicosevici, T., Prados, R., Hegedus, R., Neumann, L., de Filippo, F., Mallios, A., 2013. Mapping the Moon: using a lightweight AUV to survey the site of the 17th Century ship La Lune. In: Proceedings of MTS/IEEE OCEANS European Conference, Bergen, NO.

Hartley, R., Zisserman, A., 2003. Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge, United Kingdom.

Hildebrandt, M., Gaudig, C., Christensen, L., Natarajan, S., Paranhos, P., Albiez, J., 2012. Two years of Experiments with the AUV Dagon—A Versatile Vehicle for High Precision Visual Mapping and Algorithm Evaluation. Autonomous Underwater Vehicles (AUV), 2012 IEEE/OES.

Hogue, A., German, A., Zacher, J., Jenkin, M., 2006. Underwater 3D mapping: experiences and lessons learned. In: Proceedings of the Third Canadian Conference on Computer and Robot Vision (CRV).

Huang, A.S., Bachrach, A., Henry, P., Krainin, M., Maturana, D., Fox, D., Roy, N., 2011. Visual odometry and mapping for autonomous flight using and rgb-d camera. In: Proceedings of International Sysposium of Robotics Research (ISRR), pp. 1–16.

Ishitsuka, M., Ishii, K., 2005. Development of an underwater manipulator mounted for an AUV. In: Proceedings of MTS/IEEE OCEANS Conference, Washington, USA.

Johnson, A.E., Goldberg, S.B., Cheng, Y., Matties, L., 2008. Robust and efficient stereo feature tracking for visual odometry. In: Proceedings of IEEE ICRA, pp 39–46.

Kim, T.W., Yuh, J.K., 2004. Development of real-time control architecture for a semi-autonomous underwater vehicle for intervention missions. J. Control Eng. Pract. 12/12, 1521–1530.

Lee, D., Kim, G., Kim, D., Myung, H., Choi, H., 2010. Vision-based object detection and tracking for autonomous navigation of underwater robots. Ocean Eng. 48, 59–68.

Maki, T., Kondo, H., Ura, T., Sakamaki, T., 2008. Large-area visual mapping of an underwater vent field using the AUV Tri-Dog 1. In: Proceedings of MTS/IEEE OCEANS conference, Quebec city, Canada.

Marani, G., Choi, S.K., Yuh, J., 2009. Underwater autonomous manipulation for intervention missions AUVs. Ocean Eng. 36, 15–23.

Maurelli, F., Cartwright, J., Johnson, N., Petillot, Y., 2010. Herriot watt oceans systems lab Nessie IV autonomous underwater vehicle wins the SAUC-E competition. In: Proceedings of IEEE Conference on Mobile Robots and Competitions, Leiria, Portugal.

Mei, C., 2012. Robust and accurate pose estimation for vision-based localisation. In: Proceedings of IEEE IROS.

Prados, R., Garcia, R., Gracias, N., Escartn, J., Neumann, L., 2012. A novel blending technique for underwater giga-mosaicing. IEEE J. Ocean. Eng. 37 (4), 626–644.

Prats, M., Garcia, J.C., Wirth, S., Ribas, D., Sanz, P.J., Ridao, P., Gracias, N., Oliver, G., 2012. Multipurpose autonomous underwater intervention: a systems integration perspective. In: Proceedings of 20th Mediterranean Conference on Control and Automation (MED).

Pujol, J., 2007. The solution of nonlinear inverse problems and the levenberg-marquardt method. Geophysics 8 (72), 1–16.

Quigley, M., Conley, K., Gerkey, B.P. , Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A.Y., 2009. ROS: an Open Source Robot Operating System. In: ICRA Workshop on Open Source Software.

Ribas, D., Palomeras, N., Ridao, P., Carreras, M., Mallios, A., 2012. Girona 500 AUV: from Survey to Intervention. IEEE/ASME Tran. Mechatron. (1), 46–53.

Rigaud, V., Coste-Maniere, E., Aldon, M.J., Probert, P., Perrier, M., Rives, P., Simon, D., Lang, D., Kiener, J., Casal, A., Amar, J., Dauchez, P., Chantler, M., 1998. UNION: underwater intelligent operation and navigation. IEEE Robot. Autom. Mag. 5 (1), 25–35.

Rublee, E., Rabaud, V., Konolige, K., Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF. In: Proceedings of IEEE International Conference on Computer Vision (ICCV).

Sanz, P., Ridao, P., Oliver, G., Melchiorri, C., Casalino, G., Silvestre, C., Petilot, Y., Turetta, A., 2010. TRIDENT: a framework for autonomous underwater intervention missions with dexterous manipulation capabilities. In: Proceedings of the Seventh Symposium on Intelligent Autonomous Vehicles (IAV).

Schettini, R., Corchs, S., 2010. Underwater image processing: state of the art of restoration and image enhancement methods. EURASIP J. Adv. Signal Process., 746052.

Shortis, M.R., Seager, J.W., Williams, A., Barker, B.A., Sherlock, M., 2010. A towed body stereo-video system for deep water benthic habitat surveys. In: The Seventh Symposium on Intelligent Autonomous Vehicles (IAV).

Swain, M., Ballard, H.D., 1990. Indexing via color histograms. In: Proceedings of IEEE International Conference on Computer Vision (ICCV).

Williams, S., Pizarro, O., Mahon, I., Johnson Roberson, M., 2008. Simultaneous localization and mapping and dense stereoscopic seaoor reconstruction using an AUV. In: Proceedings of the International Symposium on Experimental Robotics.

Wilson, S.T., Sudheer, A.P., Mohan, S., 2011. Dynamic modelling, simulation and spatial control of an underwater robot equipped with a planar manipulator. In: Proceedings of International Conference on Process Automation, Control and Computing (PACC).

Wirth, S., Negre Carrasco, P.L., Oliver, G., 2013. Visual odometry for autonomous underwater vehicles. In: Proceedings of MTS/IEEE OCEANS European Conference, Bergen, NO.

Yu, S., Ura, T., Fuji, T., Kondo, H., 2001. Navigation of autonomous Underwater vehicles based on Artificial underwater landmarks. In: Proceedings of MTS/IEEE OCEANS Conference, vol. 1, Honolulu, USA, pp. 409–416.